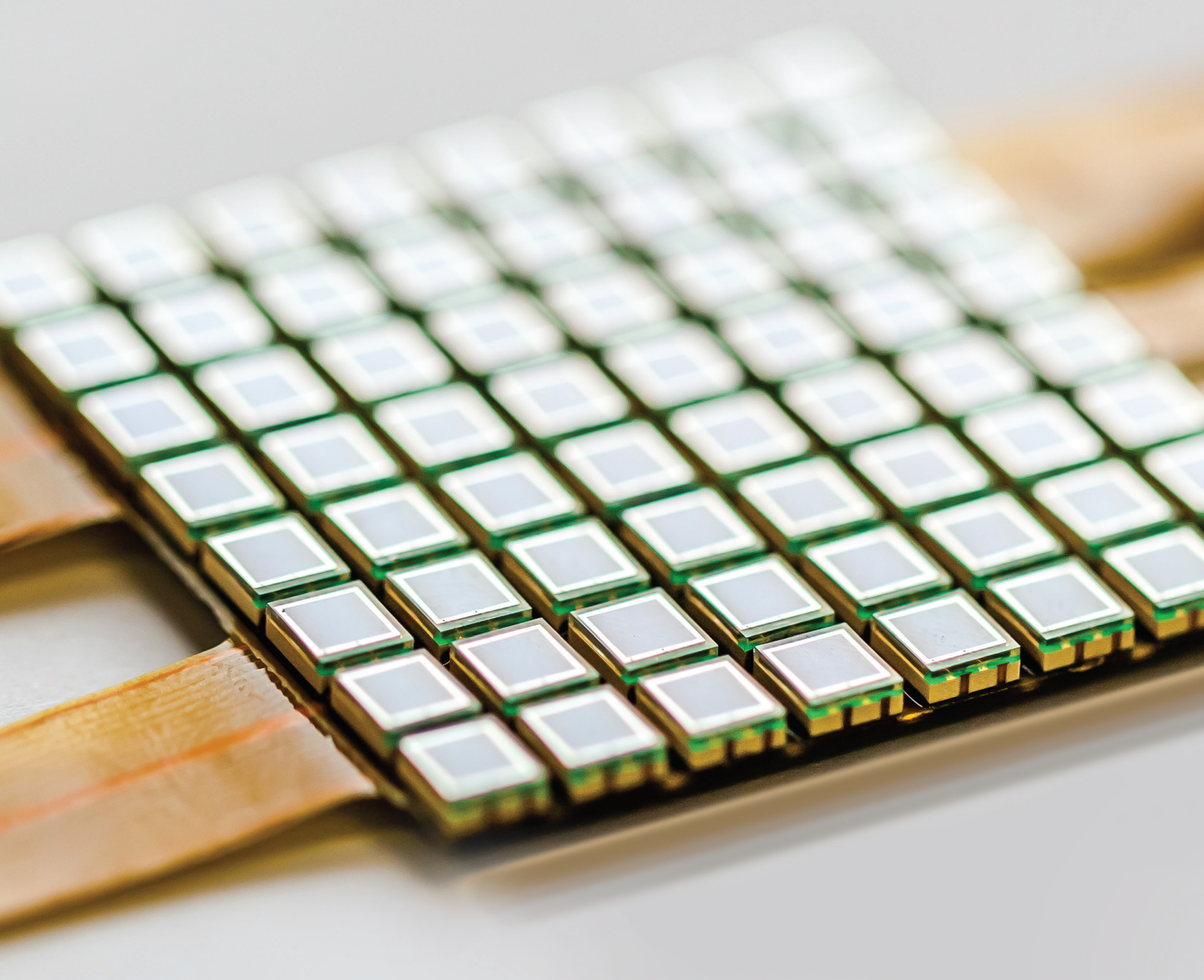# Virtual Sensors and Anomaly Detection

Lead Guest Editor: Jose Luis Calvo-Rolle
Guest Editors: Esteban Jove, Jose-Luis Casteleiro-Roca, and Dragan P. Simic

Virtual Sensors and Anomaly Detection

Virtual Sensors and Anomaly Detection

Lead Guest Editor: Jose Luis Calvo-Rolle
Guest Editors: Esteban Jove, Jose-Luis Casteleiro-Roca, and Dragan P. Simic

Chief Editor

Contents

Research Article

A Multirate Sensor Information Fusion Strategy for Multitask Fault Diagnosis Based on Convolutional Neural Network

Xiangkai Ma [iD],[1] **Pei Wang** [iD],[2] **Bozhou Zhang** [iD],[3] **and Ming Sun** [iD][2]

[1]*School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 611731, China*
[2]*School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China*
[3]*Chengdu Science and Technology Development Research Center, Chengdu 610041, China*

Correspondence should be addressed to Ming Sun; sunm@uestc.edu.cn

In complicated mechanical systems, fault diagnosis, especially regarding feature extraction from multiple sensors, remains a challenge. Most existing methods for feature extraction tend to assume that all sensors have uniform sampling rates. However, complex mechanical systems use multirate sensors. These methods use upsampling for data preprocessing to ensure that all signals at the same scale can cause certain time-frequency features to vanish. To address these issues, this paper proposes a Multirate Sensor Information Fusion Strategy (MRSIFS) for multitask fault diagnosis. The proposed method is based on multidimensional convolution blocks incorporating multisource information fusion into the convolutional neural network (CNN) architecture. Features with different sampling rates from the raw signals are run through a multichannel parallel fault feature extraction framework for fault diagnosis. Additionally, time-frequency analysis technology is used to reveal fault information in the association between time and frequency domains. The simulation platform's experimental results show that the proposed multitask model achieves higher diagnosis accuracy than the existing methods. Furthermore, manual feature selection for each task becomes unnecessary in MRSIFS, which has the potential toward a general-purpose framework.

1. Introduction

In many complicated systems, researchers tend to take the multisource data measured in the manufacturing process as the features of deep learning (DL) algorithms [1–3]. However, existing studies guarantee that all sensors operate at the same rate [4], which is often unrealistic in multisampling rate signal fusion systems. Therefore, the issue of multirate sensor information fusion is of great significance in the actual industrial environment and has received extensive attention, especially in the recent ten years [5, 6].

In recent studies, a model based on convolution takes advantage of its excellent ability to extract features from multisource signals, which has achieved excellent performance in multitask fault detection [7]. CNN can effectively extract fault features from the raw signal with its weight-sharing strategy, spatial pooling layer, local connection mechanism [8, 9], and ability in handling periodic signals. It had been proved that CNN is suitable to learn potential features

hidden in rotating mechanical signals because of its ability in handing periodic signals [10]. With the 1D-to-2D conversion of vibration signals or 1D convolutional structure, the 2D CNN models have been successfully applied in fault diagnosis directly using raw signals. In recent years, some CNN-based deep learning models have been developed for mechanical fault diagnosis.

For specific fault detection problems, some researchers have made different improvements based on the original convolutional layer, as follows: Jia et al. [11] proposed a framework called deep normalized convolutional neural network (DNCNN) for imbalanced fault classification of machinery to overcome the weakness of imbalanced distribution of machinery health conditions. Apart from extracting potential features hidden in signals, CNNs can detect local features in a deep network. Recently, Peng et al. [12] proposed a novel deeper 1D convolutional neural network (Der-1DCNN), which includes the idea of residual learning and can effectively learn high-level and abstract features

while effectively alleviating the problem of training difficulty and performance degradation of a deeper network. All these works prove that CNN is capable of mechanical data analysis. However, these works require that all signals for training and testing CNN must be acquired at the same sensor sampling frequency, which limits its further application.

Besides, a key disadvantage of 1D CNN is to detect the local correlation of signals that is deficient. In the design of the 2D CNN applied to the fault diagnosis algorithm of a multisensor fusion system, some signal preprocessing transformation procedures [13] and time-frequency analysis technology are needed. 2D CNN based on signal processing techniques (fast Fourier transform, short-time Fourier transformation, wavelet transform, etc.) has many successful applications in the field of mechanical fault diagnosis. In some works, the raw sensor time series data has been preprocessed by some methods such as frequency transformation and time-frequency transformation before being input to the 2D CNN [14, 15].

Although researchers have tried to combine multisensor signal fusion and deep learning, different sampling rates of the mechanical system's multiple components are not considered in the existing articles [16, 17]. The multirate sensor problem has become an urgent problem to be solved. However, in the previous paper [18], only the raw signals from the low-rate sensor are transferred through the upsampling network. As mentioned earlier, the existing deep learning model cannot solve the multirate sensor problem well in a complicated mechanical system with multiple components.

This paper is aimed at developing an end-to-end Multirate Sensor Information Fusion Strategy (MRSIFS), which is dedicated to improving the feature fusion of multirate sampled signals. This strategy can automatically extract sensitive fault features from multirate sensor signals for fault detection and diagnosis. Specifically, the strategy consists of three sequential phases: multirate sensor feature extraction stage, feature fusion stage, and regression stage. A key advantage of the proposed strategy is that the fault features of different scales can be automatically learned from the vibration signals of different components through parallel multichannels in complex mechanical systems. As we know, the different dimensions of the filters have different frequency resolutions. The sensitive frequency of the signal may exist in different frequency bands. The proposed multidimensional parallel convolution kernel can be used as filters with different frequency resolutions for recognition, thus effectively enhancing the frequency domain classification information of raw signals. Thus, it is effective to add a multidimensional convolution block (MDCB) in the multirate sensor feature extraction layer to extract the different scale fault classification information from the raw signals. Since the sensitive frequency bands of the multidimensional convoluted signal are included in the frequency component of the sequence signal, to combine the fault features of different scale signals, the feature fusion layer is used for cascading processing of the fault features. Besides, 1D CNN and DNN implement low sampling rate signal feature extraction and dimensionality reduction, while 2D CNN based on short-time Fourier transform (STFT) extracts higher sampling rate features [19]. The

strategy combines the advantages of the three network structures, and the three networks supplement the feature information neglected by the other side.

The proposed strategy is tested on the hydraulic system condition monitoring dataset, which is available from the UC Irvine Machine Learning Repository [20]. In the experimental part, MRSIFS are compared with the existing fault diagnosis methods in classification accuracy. The comparison results show that MRSIFS can extract fault features from multisampling sensor signals. The main contributions of our work are listed as follows:

(1) For the complicated mechanical system with multirate sampling, the designed multirate sensor feature extraction layer can extract fault features from the multirate sensor and fuse the fault feature automatically

(2) To improve the capability of the fault detection model to learn fault feature information from different frequency band signals, multidimensional convolutional blocks are used to learn rich and complementary fault information from multirate sensor signals in parallel. MRSIFS is an end-to-end deep learning model, which takes the original signal as input directly without time-consuming feature selection. Therefore, it has the potential to be extended to other industrial systems

(3) For the complicated mechanical system, the contribution of the present work is the implementation of a multitask classification framework, whereas the existing studies on the dataset of tag classification are all to propose and train a model for each task

The rest of this article is organized as follows. Section 2 discusses the related theories and the proposed framework. The experimental results and discussion are presented in Section 3. Finally, Section 4 is the conclusion.

2. Related Works

2.1. End-to-End Multisensor Model. Despite existing methods based on CNN have achieved the breakthrough performance in detection and diagnosis [21–23], there are still some shortcomings that can be improved: multirate information feature fusion can extract fault sensitive and complementary features, which are not contained in a single sensor signal, thus achieving higher accuracy and stability of the complex mechanical system [24, 25]. Nevertheless, most existing methods only used a single sensor, and few researchers attempted the multirate sensor information fusion for diagnosis. Furthermore, in previous studies about multisensor feature fusion, it is often assumed that all sensors have uniform sampling rates. In [16], the original AE signals of four independent sensor groups are first preprocessed by time-frequency analysis technology. Then, the feature matrices are converted into grey images. Finally, grey images are subsequently fed to the fine-tuned transfer learning (FTL) for fault diagnosis of different components and prediction of bearing degradation degree. In [17], the multisequence signal data collected by multiple sensors are converted into

multichannel feature matrices, and a parallel convolutional neural network (PCNN) is designed to fuse the fault information extracted from the transformed feature matrix.

Besides, the signals of various sensors are difficult to be measured at the same sampling rate, since the complicated mechanical system and design principles between different sensors [26–28]. Thus, the issue about multirate feature extraction inevitably occurs in the process of sensor fusion for complicated mechanical systems using multirate sensors [29]. However, most existing work in the area of CNN for multisensor feature fusion usage upsampling may make the vanish of certain time-frequency features, unless all sample rates are sufficiently close, and thus degrades the accuracy of fault diagnosis. In the multisensor system with different sampling rates, Li et al. [18] present an improved information fusion framework based on the atrous convolution. Specifically, to avoid tedious preprocessing, the model extracts fault features from multisource signals by constructing a convolution kernel of adaptive size matching the data source channels. Finally, the proposed method is compared with the existing research, as shown in Table 1.

2.2. Existing Works Based on the Same Dataset. There have been some researches about the hydraulic system condition monitoring dataset based on time-frequency analysis techniques and artificial feature extraction. Helwig et al. [31, 32] convert the time domain data into frequency domain using fast Fourier transform and generate statistical features. They then calculated features for fault label correlation and selected the n features by ranking or sorting the correlation (CS) for the fault classification. Prakash and Kankar [33] also utilized statistical features of frequency domain data and applied XGBoost [34] to define feature importance (XFI) and select half of the highest correlations along with a deep neural network for the classification model.

These previous approaches include several drawbacks. First, the proposed handcrafted feature selection methods, such as CS and XFI, may suffer from utilizing redundant features, which will disrupt the learning of the model. Second, statistical feature extraction, PCA, and other feature engineering methods are not suitable for real-time detection. These techniques must be applied to each new incoming sample, thereby consuming more time and computation power. Third, to ensure the quality of the feature extraction, a manual design and suitable features are needed based on the characteristics of the different types of faults. Furthermore, feature extraction usually turns out to be a computationally costly operation, but the existing studies on the dataset of tag classification based on UCI are all to propose and train a model for each tag. The quality of the features directly determines the system performance. Therefore, the system feature extraction is not automatic.

With the aforementioned open issues, we propose the end-to-end deep learning model and directly take the raw vibration signals as input. Finally, the multilabel classification problem is transformed into the regression prediction problem, and the accuracy rate is higher than the existing methods [31–34]. Unlike traditional methods relying on manually defined or extracted features, the proposed design does not require any additional expert knowledge, which has great potential toward a general-purpose framework for intelligent fault diagnosis. Thus, it can be easily extended to deal with fault diagnosis problems of other industrial systems.

3. Methods

Although the most satisfactory level of anomaly diagnosis accuracy was reported [20, 32–34], most of these previous approaches had to design and train different feature extraction models for each task. In contrast, the multitask model is suitable for the complex mechanical system. In addition, manual feature selection often results in wasteful computational costs, but the existing studies on the dataset of tag classification based on UCI are all to propose and train a model for each tag. However, the stage of manual feature selection is difficult and time-consuming [35]. This process incurs considerable computational costs, which eventually may impede the use of existing approaches in real-time fault diagnosis applications. The MRSIFS model proposed in this paper transforms the multirate problem into a unified multidimensional convolutional neural network feature fusion strategy. Compared with the traditional methods about condition monitoring of hydraulic system dataset, MRSIFS has better robustness in noise environment and does not require predetermined feature selection. In particular, it proposes a feature extraction layer based on a multidimensional convolutional neural network and a concatenated feature fusion layer. Finally, the multitask classification problem is transformed into the regression prediction problem, and the accuracy rate is higher than the existing methods [20, 32–34].

The framework of the proposed MRSIFS is shown in Figure 1. The inputs to the model consist of 3 segments of raw temporal vibration signal at different sampling rates. The output Y is a vector made up of the labels of five tasks, where each value represents the predicted regression value for a task. The fault detection is defined to predict the vector Y based on the fault feature extraction from the raw temporal vibration signal X using the MRSIFS model. The MRSIFS model has three parts: the multidimensional convolution feature learning layer, the multirate feature concatenate layer, and the regression layer.

3.1. The Framework of MRSIFS. In the multirate sensor feature extraction stage, the fault feature extracted by the input layer is fed into the multidimensional convolution feature learning block. The kernels of the convolution layer have different dimensions. As we know, convolution kernels of different dimensions act as filters of different resolution scales to extract fault features in the raw signals and simultaneously extract the features of input signals in different frequency bands. As shown in Figure 2, features of the convolution kernel dimension are combined through the concatenation layer to form a multirate feature map. As a fault information collector, the concatenation layer can aggregate features of different scales to form a multirate feature set. It can be observed that the data from multiple sensors are transformed into multiple channels through the input layer, and then, the fault features are obtained through the feature

TABLE 1: Comparison with related works.

Framework	MSFTFI [16]	PCNN [17]	FAC-CNN [18]	MRSIFS
Data preprocessing	STFT	——	——	STFT
CNN dimension	2D	2D	1D	1D, 2D
Feature extraction	FTL, LeNet-5	LSTM, DRN	Atrous CNN	FPN [30]
Activation function	ReLU	ReLU	PReLU	PReLU
Sampling frequency	Fixed	Fixed	Different	Different
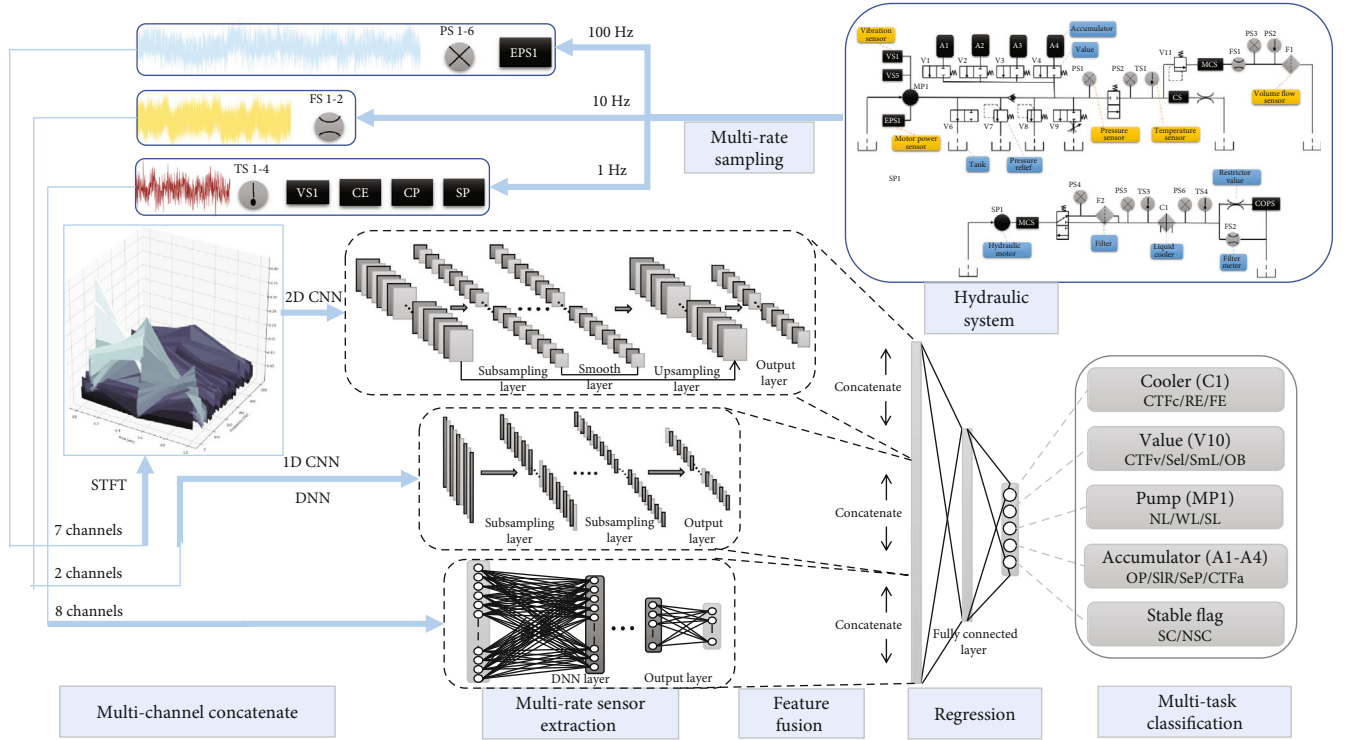
FIGURE 1: Flowchart of the proposed approach.

extraction layer and feature fusion layer sequentially. Specifically, the input layer of the proposed method is a three-dimensional matrix, and the prediction results are taken as the output of the multitask model. The specific size and other parameters of the filter are shown in Tables 2–4.

3.2. CNN Details and Regression. The 2D CNN is composed of six parts, including STFT, input layer, downsampling layer, smooth layer, and upsampling layer. The downsampling layer is composed of K prevention blocks, and each of them contains three convolutional layers. The upsampling layer is implemented based on bilinear. And the smooth layer contains a single convolutional layer. To extract the hidden fault information in the signal as far as possible, four downsampling layers, four smooth layers, and three upsampling layers were successively used in the experimental model with the dataset. The features are fed into the output layer and straighten the output of the last layer as the fault feature extracted by the 2D convolution component. The 2D CNN model summary is mentioned in Table 2. Table 2 describes the type of layers, in channels, out channels, kernel size, etc.

It must be noted that in the 2D convolution component, we use a deep convolutional network structure. In this way, although deep-level fault features can be extracted, according to some existing studies [30], the deep-level convolutional network may lose some shallow-level fault features. In this paper, the downsampling layer, smooth layer, and upsampling layer are used to cooperate, so that the features extracted from the shallow convolutional network can be integrated into the subsequent deep-level feature information. The proposed network structure is inspired by [36].

Compared with the two-dimensional convolution structure, the one-dimensional convolution structure only retains the part of the lower sampling layer, and some modifications are made in the convolutional network parameter settings. The DNN network block is composed of four full connection layers. Similarly, the regression layer consists of three full connection layers, with ReLU as an activation function. Parameter details and design implementation of DNN and regression layer are shown in Tables 3 and 4, respectively. After each convolutional layer in the input layer and downsampling layer, and the fully connected layer in DNN, batch
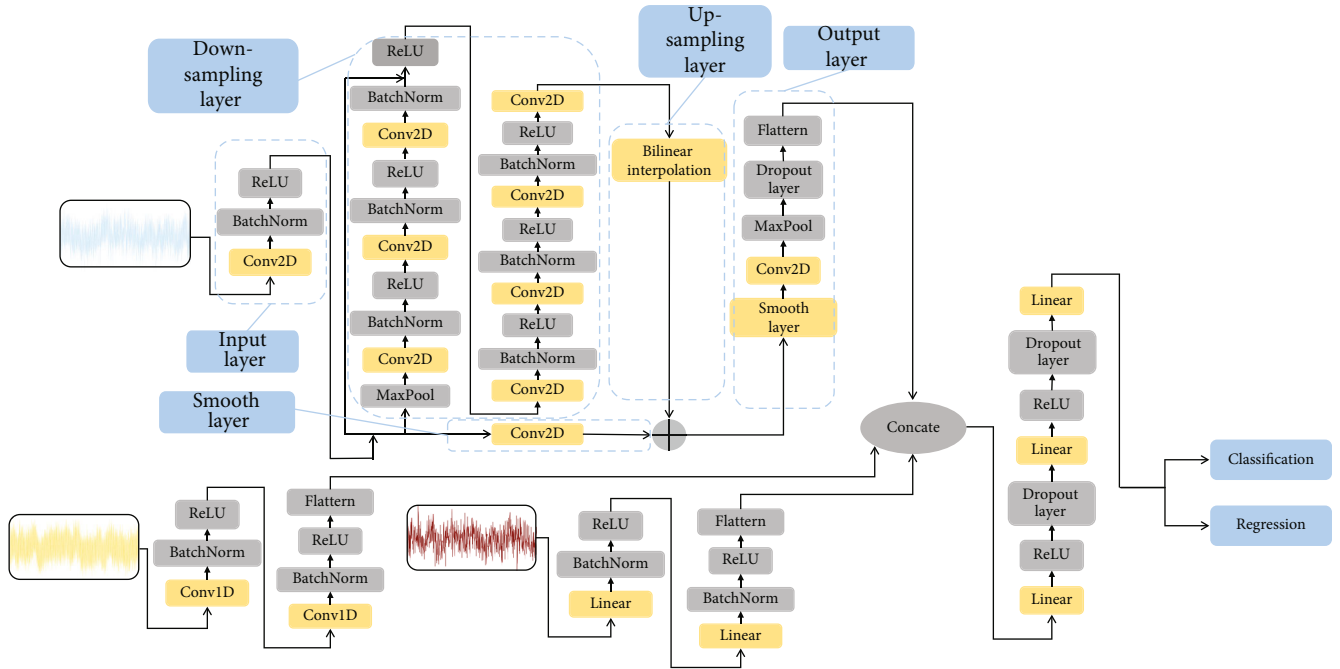
FIGURE 2: Detailed structure of the proposed method.

TABLE 2: Summary of the submodel of MRSIFS using 2D CNN for feature extraction.

Network structure	Layers	In_channels	Out_channels	Kernel_size	Stride	Padding
Input layer	Conv2D	7	16	7	2	3
	MaxPool			3	2	1
	Conv2D	16	16	1	1	0
	Conv2D	16	16	3	1	1
Downsampling layer	Conv2D	16	64	1	1	0
	Conv2D	64	16	1	1	0
	Conv2D	16	16	3	1	1
	Conv2D	16	64	1	1	0
Smooth layer	Conv2D	64	64	1	1	0
Upsampling layer	"Bilinear"					
	Conv2D	64	32	1	1	0
Output layer	MaxPool			4	4	0
	Dropout ($p = 0.2$)					

TABLE 3: Summary of the submodel of MRSIFS using DNN for feature extraction.

Network structure	Layers	In_features	Out_features	Bias
	Linear	480	400	True
Fully connected layer	Linear	400	256	True
	Linear	256	128	True
	Linear	128	64	True

TABLE 4: Summary of the regression layer of MRSIFS.

Network structure	Layers	In_features	Out_features	Bias
	Linear	1280	512	True
	Dropout ($p = 0.2$)			
Regression	Linear	512	128	True
	Dropout ($p = 0.2$)			
	Linear	128	5	True

normalization [37] is used to accelerate the training process of MRSIFS.

3.3. Training Details. In MRSIFS training, we adopt the mean-square error (MSE) between the predictive label and the real label as the loss function. To reduce the computational overhead in the training, the Adam algorithm [38] is employed to optimize the parameters of the model. A critical task for MRSIFS training is the adjustment of hyperparameters, and this paper takes the batch size and learning rate

as hyperparameters. The batch size defines the number of samples to be processed in one batch. The learning rate determines the convergence speed of weight and bias in the neural network during training. The learning rate of model training is set to 0.001, and the batch size is set to 32. To prevent overfitting, the proposed method uses dropout technology [39] between the full connection layers in the regression prediction stage. With the dropout technique, every parameter in the full connection layer has a certain probability of being randomly removed during each training epoch. In the initialization stage of the model, all parameters of the neural network are initialized by a zero-mean standard uniform distribution. Particularly, in this model, the biases of all neurons are set to zero when initialized. In addition, all the original vibration signals are randomly used for training the model and testing the model. Specifically, 20% of the signals are selected as test samples, and the remaining 80% are used for testing models. To obtain a relatively stable experimental result, 10 trials of MRSIFS were repeated on the condition that each group of models had the same parameters.

Since its different scales and different depths of feature extraction layers (i.e., pairs of downsampling layers, smooth layer, and upsampling layers), MRSIFS architecture has the advantages of general purpose and flexibility. Furthermore, MRSIFS can effectively learn sensitive diagnostic information by using multidimensional feature extraction structure and capture complementary and useful fault features at different scales for fault diagnosis and detection. As we know, more robust and abstract fault information is expected to improve the diagnosis performance. Therefore, MRSIFS with more scales and layers can extract useful diagnostic features to adapt to the complicated mechanical system. In addition, in practical applications, the simple rate of the input samples and the depth of the layers have a restriction relationship. In MRSIFS, researchers can select the appropriate kernel size and layer depth based on the length of the input signal. More details can be found in III.

The proposed MRSIFS algorithm is conducted on torch 1.6.0+cu101. The hardware configuration for training and testing is Intel(R) MKL-DNN v1.5.0 + TITAN X (Pascal), while the software environment is Linux + Python 3.7 Version 3.7.9 [GCC 7.3.0].

4. Experiments and Discussion

4.1. Data Description. The proposed strategy is tested on the hydraulic system condition monitoring dataset, which is available from the UC Irvine Machine Learning Repository [10]. The system cyclically repeats constant load cycles (duration 60 seconds) and measures process values such as pressure, volume, flow, and temperature, while the conditions of the four hydraulic components (cooler, valve, pump, and accumulator) and the stable flag of the hydraulic system are quantitatively varied. As shown in Table 5, the dataset consists of the measurement signals of 17 sensors, including 14 physical sensor components and 3 virtual sensors. The measurement period of each sensor is 60 seconds, and the sampling rate range is between 1 Hz and 100 Hz. The dataset consists of 2205 samples with a sampling period of 60

Table 5: Hydraulic system sensors. *CE, CP, and SE are virtual sensors.

Sensor	Physical quantity description	Unit	Sampling rate	Sensor signal length
PS1-6	Pressure	Bar	100 Hz	6000
EPS1	Motor power	W	100 Hz	6000
FS1-2	Volume flow	L/min	10 Hz	600
TS1-4	Temperature	°C	1 Hz	60
VS1	Vibration	mm/s	1 Hz	60
CE (virtual)	Cooling efficiency	%	1 Hz	60
CP (virtual)	Cooling power	kW	1 Hz	60
SE (virtual)	Efficiency factor	%	1 Hz	60

seconds. Each sample contains a component status label that reflects the fault condition of the 5 components.

As shown in Table 6, the training results of each sample include five types of tags. Therefore, what this paper deals with is a multitask classification problem. Nevertheless, because the value of each tag has a specific physical meaning, therefore, the multitask classification problem can also be transformed into a regression problem to solve. The problem proposed by the dataset is actually using the regression vector with the predicted length of 5, and to set the threshold, convert the regression value to the final classification result. Ideally, the output of the network for any input should be limited to our given range (for example, the labelled hydraulic accumulator is estimated to be only 130, 115, 100, and 90), but actually, the output value is often in between these values. In this paper, the value with the minimum absolute value error is selected as the predicted class label.

4.2. Compared Models. The purpose of the experiment is to verify the higher classification accuracy of MRSIFS proposed in this paper compared with the existing MSFTFI, PCNN, and FAC-CNN. Besides, when changing the convolutional layer depth of the MRSIFS feature extraction block, MRSIFS still has good performance, which indicates that MRSIFS is not sensitive to the setting of super parameters, indicating that the strategy proposed in this paper has wide adaptability. The main comparisons of the proposed model and existing methods are listed as follows:

(1) MSFTFI: in [16], the original AE signals of four independent sensor groups are first preprocessed by time-frequency analysis technology. Then, the feature matrices are converted into grey images. Finally, grey images are subsequently fed to the fine-tuned transfer learning (FTL) for fault diagnosis of different components and prediction of bearing degradation degree

(2) PCNN: firstly, the multisequence signal data collected by multiple sensors are converted into multichannel feature matrices, and a parallel CNN is designed to concatenate the fault information extracted from the

TABLE 6: Summary of components and their simulated fault conditions of the hydraulic dataset. A load cycle with 60 s duration is repeated 2205 times with the distribution of instances as indicated by the examples.

Component	Fault conditions	States	Abbreviation	Examples
Cooler (C1)	Cooling power decrease	3%: close to total failure	CTFc	732
		20%: reduced efficiency	RE	732
		100%: full efficiency	FE	741
Valve (V10)	Switching degradation	73%: close to total failure	CTFv	360
		80%: severe lag	SeL	360
		90%: small lag	SmL	360
		100%: optimal behaviour	OB	1125
Pump (MP1)	Internal leakage	0: no leakage	NL	1221
		1: weak leakage	WL	492
		2: severe leakage	SL	492
Accumulator (A1-A4)	Gas leakage	130 bar: optimal pressure	OP	599
		115 bar: slightly reduced	SlR	399
		100 bar: severely reduced	SeP	399
		90 bar: close to total failure	CTFa	80 8
Stable flag (S1)	——	0: conditions were stable	SC	1449
		1: nonstatic conditions	NSC	756

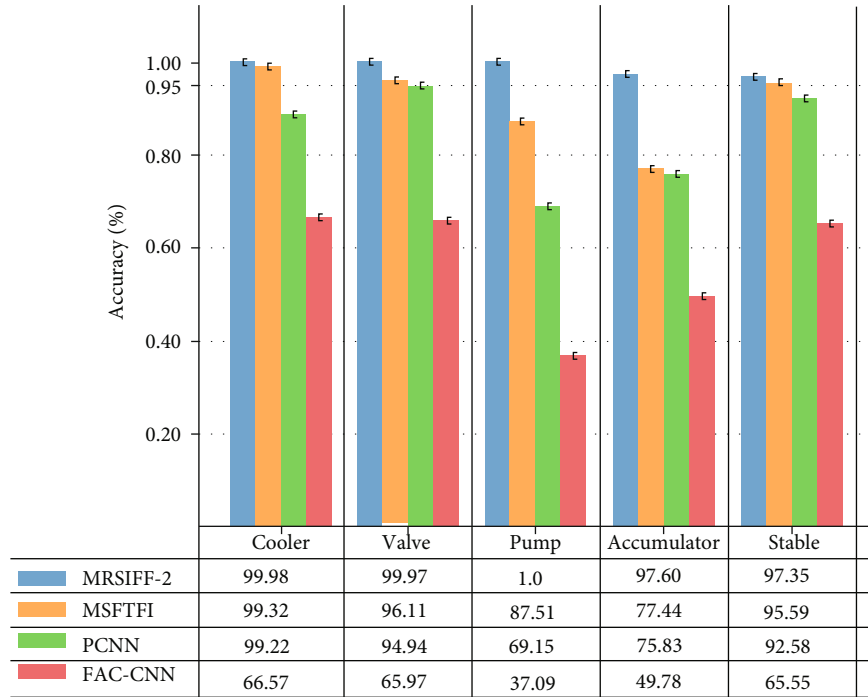transformed feature matrix. More details of PCNN can be found in [17]

(3) FA-CNN: researchers construct a convolution kernel of adaptive size matching data source channel to capture multiscale data without time-frequency analysis technology. In addition, to extract the diagnosis information of the fused data effectively, one-dimensional CNN and global average pooling methods are adopted to improve the domain adaptation of the network. More details of FA-CNN can be found in [18]

(4) MRSIFS: in comparison experiments, MRSIFS were tested on layers 1 to 4 to test the depth of influence of MRSIFS on diagnostic performance. In addition, to provide a fair comparison, all comparison models have the same model depth as the MRSIFS proposed

To enable the fairness of comparison and improve the persuasiveness of the experiment results, all existing models have the same parameters and structure as the proposed MRSIFS. In addition, the same input form, number of training, batch size, and parameter optimization algorithm were adopted for all models.
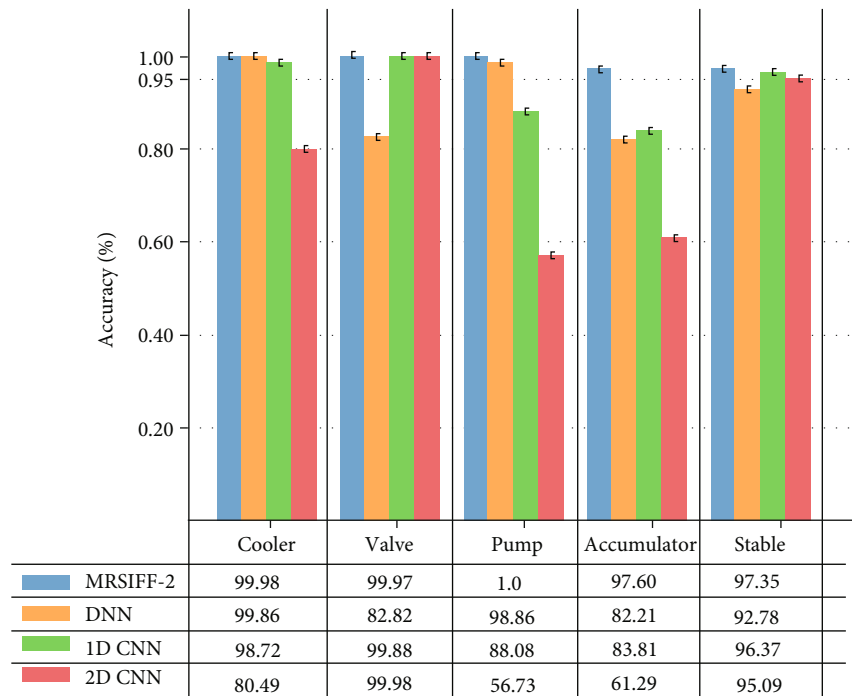
4.3. Diagnosis Results and Performance Comparison. To reduce the effect of randomness and improve the persuasiveness of experiment results, the average accuracies of ten replicate experiments for multitask fault diagnosis using different models are shown in Figure 3(a). The standard deviations of the cumulative experiments act as the error bars, which reflect the robustness of the method. In the figure, the mean represents the average accuracy of the model when solving the assigned tag task, and the standard deviation reflects the stability of the model.

As shown in Figure 3(a), the proposed MRSIFS achieves optimum performance on five tasks (coolers, valves, pumps, accumulators, and stabilizers). Specifically, the fault diagnosis accuracies of MRSIFS for each label are more than 97%, so MRSIFS has good multirate information extraction capability on multitask classification. Besides, since the standard deviations of MRSIFS on all tasks are very small, the proposed method has better robustness. The MSFTFI, PCNN, and FAC-CNN perform slightly less well than MRSIFS in the tag cooler, valve, and pump but have significantly worse performed than MRSIFS in tag accumulator and stable. Besides, as shown in Figure 3(b), MRSIFS perform better than the original CNNs and DNN in all five tasks. Therefore, we can draw two advantages, the first advantage is that MRSIFS performs better than the original CNNs. Another advantage of the proposed MRSIFS is that using multidimensional convolution kernels can extract more useful multiscale fault information than the traditional multisensor feature extraction model. Additionally, the multiscale feature connection layer fuses the fault feature information extracted from the original signals of different sampling frequencies by each submodel. Although FAC-CNN, PCNN, and MSFTFI can achieve excellent performance in task cooler, valve, and pump, they have lower than 80% accuracy in other tasks, which show that the existing models have low multirate feature extraction ability. The convergence performances of four different models are shown in Figure 3(b). We can see that the proposed MRSIFS converges faster than other models, which can be concluded that multidimensional cascade signals can provide more specific information than the raw signals.

To reveal that the existing models lack sufficient feature extraction capability, the losses of the four models on the condition monitoring of the hydraulic system dataset are shown in Figure 4. First of all, the loss of MRSIFS-2 (the depth of the multidimensional convolution feature learning

		Cooler	Valve	Pump	Accumulator	Stable
■	MRSIFF-2	99.98	99.97	1.0	97.60	97.35
■	MSFTFI	99.32	96.11	87.51	77.44	95.59
■	PCNN	99.22	94.94	69.15	75.83	92.58
■	FAC-CNN	66.57	65.97	37.09	49.78	65.55

(a)

		Cooler	Valve	Pump	Accumulator	Stable
■	MRSIFF-2	99.98	99.97	1.0	97.60	97.35
■	DNN	99.86	82.82	98.86	82.21	92.78
■	1D CNN	98.72	99.88	88.08	83.81	96.37
■	2D CNN	80.49	99.98	56.73	61.29	95.09

(b)

FIGURE 3: Testing fault diagnosis accuracy about four models (a) and the performance of the proposed MRSIFS in five tasks compared to all submodels (b).

layer is two) converges to zero. However, the loss of PCNN converges to about 0.1. Secondly, when more than 80 epochs were trained, the loss of MRSIFS-2 converges to zero, whereas the MSFTFI model needs to train more than 110 epochs to obtain similar results. In addition, since all labels of the raw signals have been normalized and preprocessed,

the losses of MRSIFS-2 and FAC-CNN are highly overlapping. To further explore the effectiveness of the proposed MRSIFS framework, the logarithmic function can be introduced to enlarge the difference between MRSIFS-2 and FAC-CNN. The logarithmic function value of the loss of MRSIFS is one order of magnitude lower than that of the
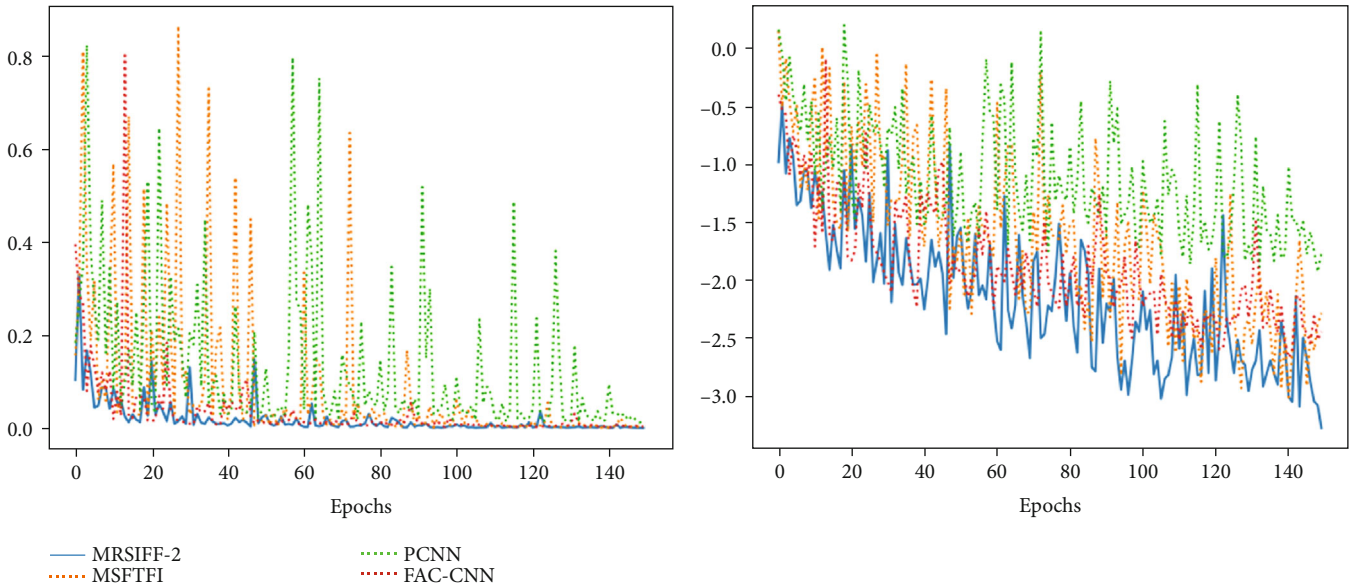
FIGURE 4: The losses of the four models based on the condition monitoring of the hydraulic system dataset.
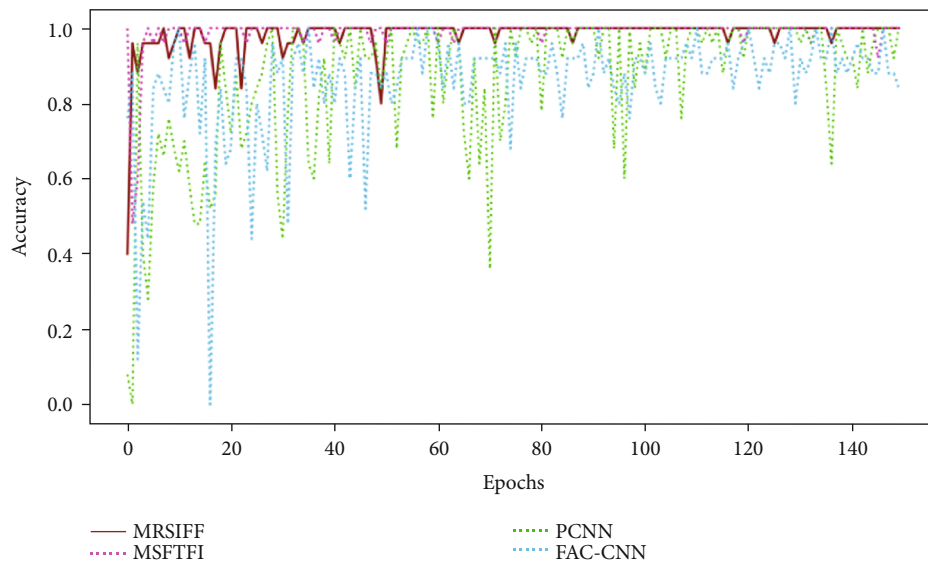

FIGURE 5: Training performance curve in terms of overall accuracy.

three existing models, which illustrates that the proposed MRSIFS model can learn the fault features and diagnosis information robustly from the raw vibration signals, and has fault discriminative ability. Thus, it can be proved that the multidimensional convolution feature learning stage actually makes the diagnosis information of the signal complementary by concatenating the useful components that come from multirate sensor systems.

It shows the overall accuracy curves for both MRSIFS, MSFTFI, PCNN, and FAC-CNN over 150 epochs in Figure 5 during the entire training process. As shown in Figure 5, compared with existing models, the proposed MRSIFS has stabilized over 95 accuracies after training 50 epochs. MRSIFS is more adaptable to complicated mechanical systems since it requires less time for training.

4.4. Feature Visualization via t-SNE and Confusion Matrix. To further prove that multidimensional feature learning can improve the feature extraction capability of the model in complicated mechanical systems, the t-SNE method [40] is adopted to realize the visualization of the feature maps learned from the multidimensional convolution block. The feature map obtained through the feature extraction stage of multidimensional convolution is shown in Figure 6, which uses different colours to distinguish the features of samples with different tag types. Only consider the classification results of the original data of the four models when the hydraulic accumulator is used as the label because when the other four hydraulic components are used as the label, the accuracy of the classification results obtained by the MRSIFS model has reached 99.5%.

MRSIFF

● 130: optimal pressure
● 115: slightly reduced pressure
● 100: severely reduced pressure
● 90: close to total pressure

(a) MRSIFF

MSFTFI

● 130: optimal pressure
● 115: slightly reduced pressure
● 100: severely reduced pressure
● 90: close to total pressure

(b) MSFTFI

FAC-CNN

● 130: optimal pressure
● 115: slightly reduced pressure
● 100: severely reduced pressure
● 90: close to total pressure

(c) FAC-CNN

PCNN

● 130: optimal pressure
● 115: slightly reduced pressure
● 100: severely reduced pressure
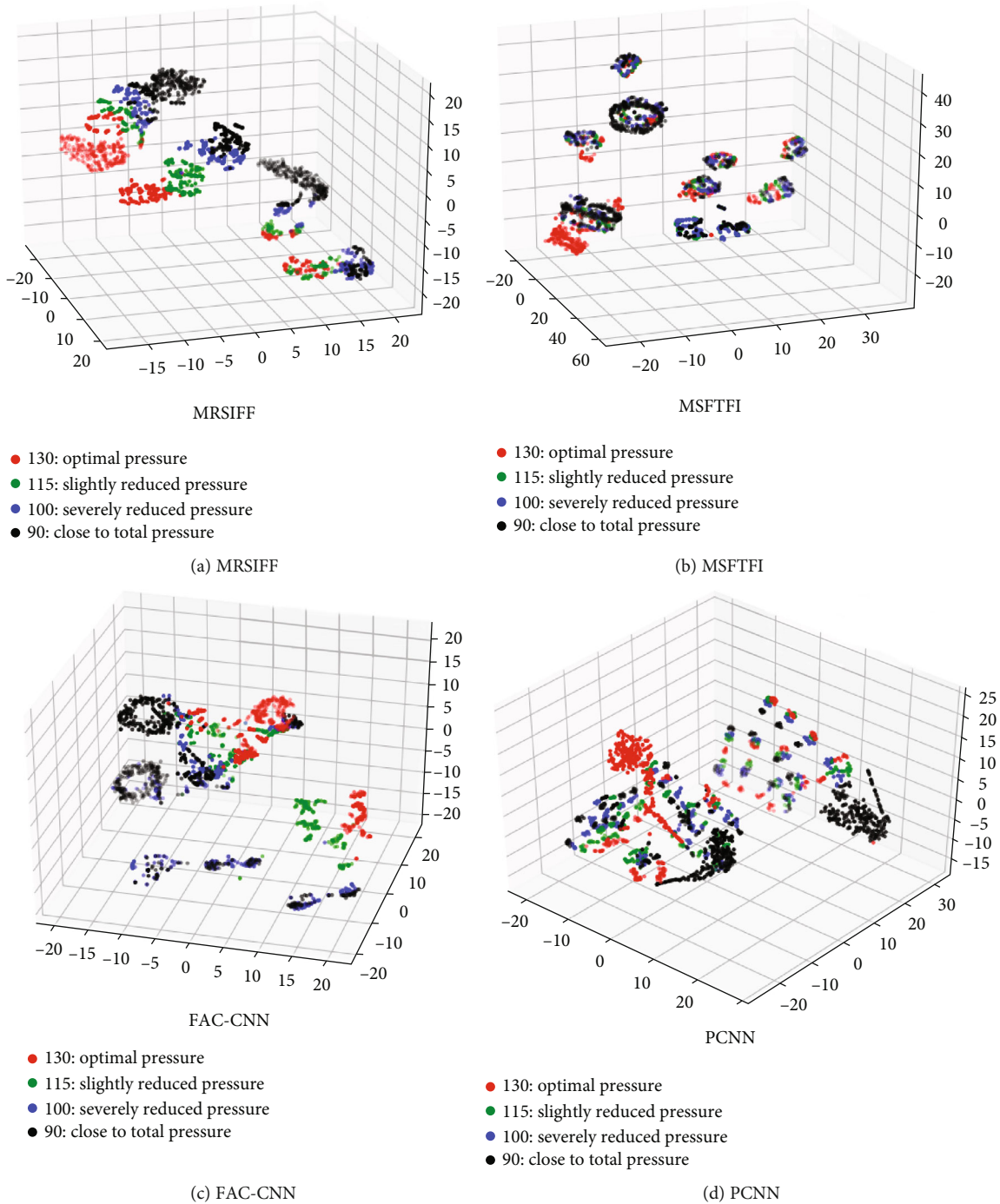● 90: close to total pressure

(d) PCNN

FIGURE 6: The feature visualization of different models.

Specifically, for the multitask classification problem, t-SNE is used to display the classification effect of the model better. As can be seen from Figure 6(a), samples are grouped into four categories under other labels, and each category is further divided into four categories, with accumulator samples as a label. However, in Figures 6(b)–6(d), the clustering results of the features of samples with the same tag type are worse, which indicates that the existing models cannot extract fault features from a multirate sensor. On the contrary, the proposed MRSIFS model has good domain adaptability and diagnosis information extraction ability

under complex working conditions, so it can finally learn fault discriminative features robustly.

Figure 7 gives the probability of the model making the correct classification and the probability of making the wrong classification for a given health condition. The X-axis represents the accurate label, and the Y-axis represents the predicted label.

4.5. Time Consumption. It is worth noting that, to make the comparison results more convincing, all the existing models and the proposed MRSIFS have the same convolutional layer
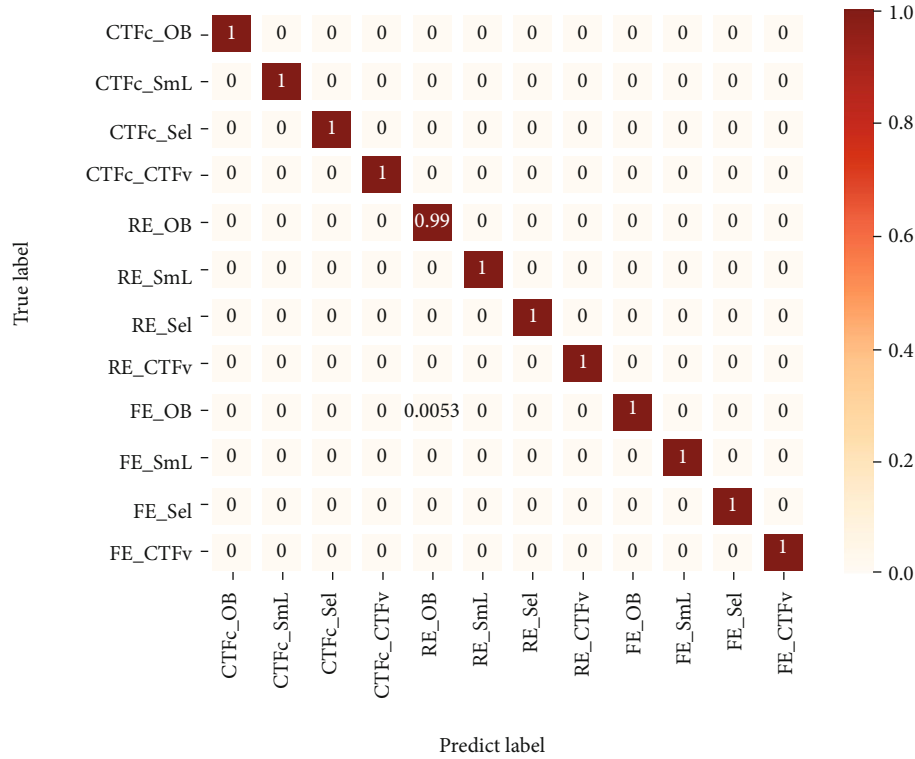
FIGURE 7: Confusion matrix results of cooler and valve.

depth. The training time and testing time spent by different models on the hydraulic system dataset are shown in Table 7. As shown in Table 7, MRSIFS, MSFTFI, and PCNN all require more time consuming than the original CNNs, which can be explained by the introduction of more parameters in multidimensional convolution blocks and result in more computing time. The testing time is a determining factor in the performance of the online fault diagnosis and detection model since the model is trained offline. Even if MRSIFS-2 needs more time to test, the testing time of the proposed method on the hydraulic system dataset is just 14.2161 ms, which shows the possibility about using MRSIFS-2 for online fault detection and diagnosis.

4.6. Discussions on the Effects of Depth. The depth of the convolutional layer will affect the diagnostic performance of the model based on deep learning. The proposed MRSIFS model can automatically adjust the depth of the model (i.e., the number of convolutional layer and pool layer pairs) according to the characteristics of the dataset. This section explores the influence of depth on feature extraction and fault diagnosis. The abstraction level of the feature is determined by the depth of MRSIFS and the scale of the convolution kernel. In complicated mechanical systems, the accuracies of classification results depend largely on the abstraction level of fault features. To test the influence of model depth on fault diagnosis, the accuracy of MRSIFS with one to four layers is recorded and compared with existing models in Figure 8.

As we know, the abstraction level of the extracted features can be determined by the depth of the MRSIFS. In addition, speed or load variations and background noise may suffer

TABLE 7: Cost time for MRSIFS for different scales and other models.

Method	Training time (s)/epoch	Testing time (ms)/one sample
MRSIFS-1	1.7278	2.1811
MRSIFS-2	2.0080	2.8432
MRSIFS-3	2.0813	3.7681
MRSIFS-4	2.2856	4.4364
MSFTFI	2.0809	1.7704
PCNN	1.7067	2.5053
FAC-CNN	1.4600	1.1219
DNN	1.3744	0.8290
1D CNN	1.5845	1.7763
2D CNN	1.7792	1.8995

low-level features obtained from the raw signal. Since the abstraction level of fault features can significantly impact the classification results, MRSIFS with layer depth of 1 to 4 are tested in this paper to investigate the influence of depth on diagnostic performance. The result is shown in Figure 8. In these experiments, the average accuracies of ten replicate experiments for each condition are shown in Figure 8, where the standard deviations of the cumulative experiments act as the error bars, which reflect the robustness of the method. A conclusion from the figure is that, for each tag, all MRSIFS outperform MSFTFI, PCNN, and FA-CNN in the fault diagnosis accuracy index. In general, as the depth increases to equal to two, MRSIFS can achieve the best and most reliable

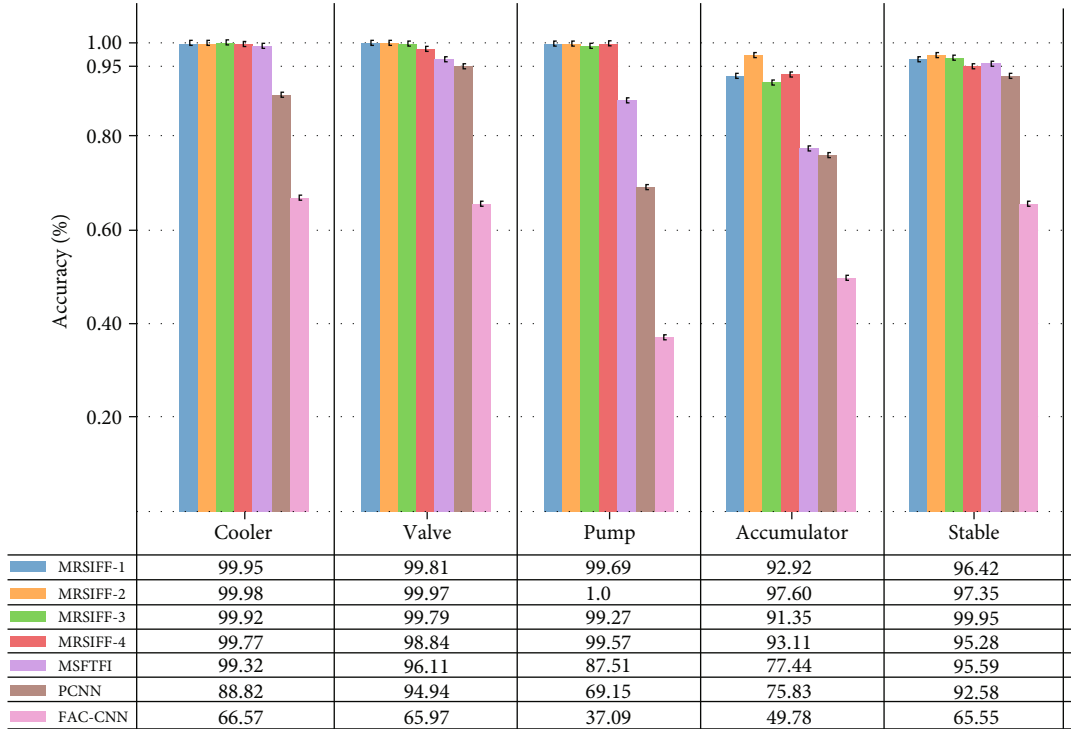|	Cooler	Valve	Pump	Accumulator	Stable
MRSIFF-1	99.95	99.81	99.69	92.92	96.42
MRSIFF-2	99.98	99.97	1.0	97.60	97.35
MRSIFF-3	99.92	99.79	99.27	91.35	99.95
MRSIFF-4	99.77	98.84	99.57	93.11	95.28
MSFTFI	99.32	96.11	87.51	77.44	95.59
PCNN	88.82	94.94	69.15	75.83	92.58
FAC-CNN	66.57	65.97	37.09	49.78	65.55

FIGURE 8: Diagnosis performance on the testing set using the MRSIFS with different scales from 1 to 4.

performance. Specifically, for both MRSIFS, MSFTFI, PCNN, and FAC-CNN, since MRSIFS with multidimensional convolution kernels can extract more useful and abstract fault information, which help in classification at higher levels, the classification performance improves with increasing depth. As the depth continues increasing, the accuracy of MRSIFS begins to decline. In addition, the standard deviation of each condition is smaller than these existing models, especially MRSIFS in the fourth task, which showed more excellent fault diagnosis performance than existing literatures. The result proves that MRSIFS can learn more abstract and fault sensitive features from multirate sensor systems. The time consumption of MRSIFS for training and testing at different convolutional layer depths was calculated, and the results are shown in Table 7. Obviously, the time overhead for optimizing model parameters increases with increasing depth. Therefore, to reduce the computational cost, MRSIFS with two layers are selected for feature extraction. For complex diagnostic tasks in practical applications, MRSIFS can be modified to improve the performance further.

Another advantage of the proposed model is that, for a compound and changeable industrial environment, MRSIFS does not require a complex parameter adjustment process and can effectively extract fault information. Therefore, the model introduced in this paper is not sensitive to the setting of neural network parameters.

4.7. The Details of MRSIFF. To reveal the function of a multidimensional convolution feature learning layer, the evolution of the neurons in the MRSIFF, the learned kernels, multirate sensor cascade signals, and the output of each layer in the MRSIFF are displayed in this section. It can be seen

from Figure 9 that the length of multiscale feature fusion signals has been increased because of the series connection of signals convoluted by kernels with different dimensions. In addition, the waveform shapes of the fusion signals have been changed. In Figure 9(a), there is little difference between the signal wave shapes (SWS) of ball and normal. However, in Figure 9(b), the fusion-SWS of them are totally different, which can provide more useful features for the classifier.

To reveal how the raw signals change in the MRSIFF and what is the input of the classifier, the evolution of the inputs in MRSIFF and the features of Multilayer Perceptron (MLP) are shown in Figure 10, respectively. It can be seen from Figure 10 that the outputs of the C3 (the third convolutional layer) are similar to the wave shape of fusion signals and they can be distinguished obviously in the four conditions. The MLP features are constituted by the features in C3, and it can be seen from Figure 10 that the features before putting into the classifier are linear separable, which can increase the classification accuracy. Furthermore, the fully connected features of MRSIFF, MSFTFI, PCNN, and FAC-CNN are mapped into three-dimension features using t-SNE which are shown in Figure 6. Obviously, the mapped features of the MRSIFF cluster are better than other models based on CNN even though other models can differentiate most samples. Thus, the t-SNE error of MRSIFF is the lowest among the three models.

4.8. Summary of Experimental Results. In this paper, another deep convolution structure is proposed, which is different from the traditional CNN structure that only utilizes features in the last convolution layer. In the feature extraction stage, the upsampling layer is combined with the

Sampling point OP

Sampling point SlR

Sampling point SeP

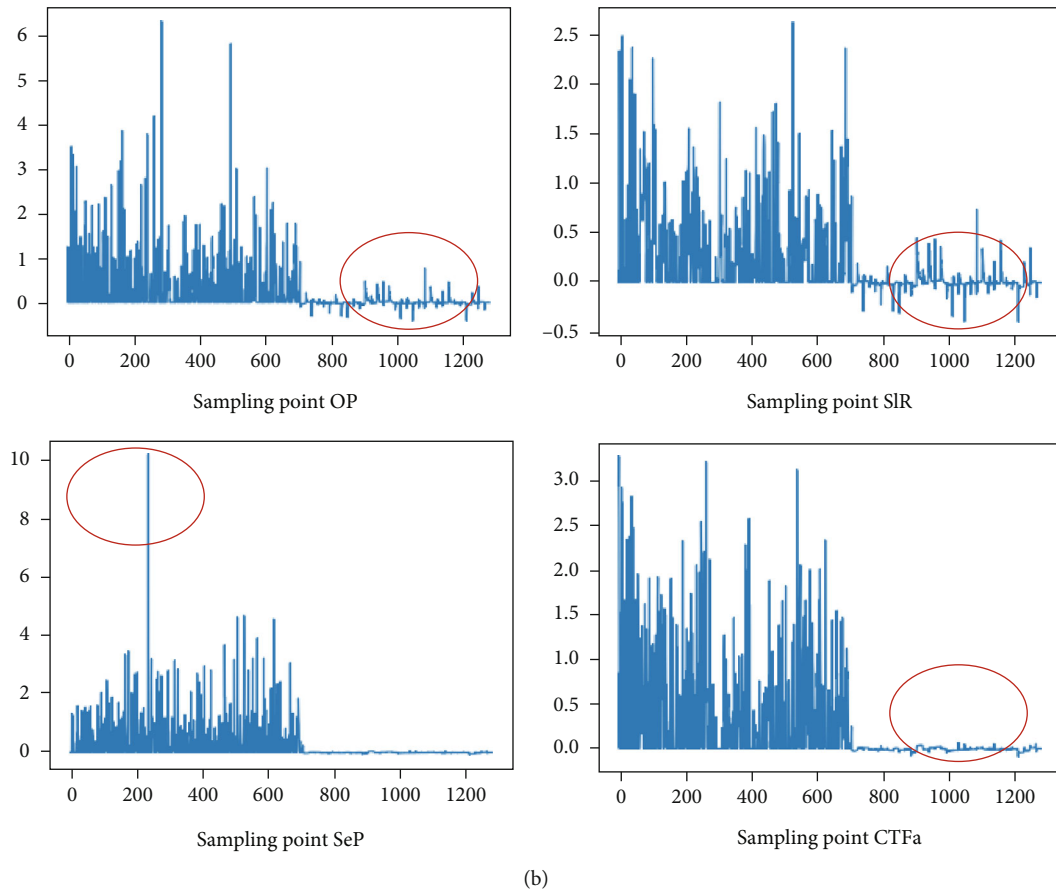Sampling point CTFa

(a)

Figure 9: Continued.

(b)

FIGURE 9: Raw signal (a) and fusion signal (b) of each healthy condition of the hydraulic system.
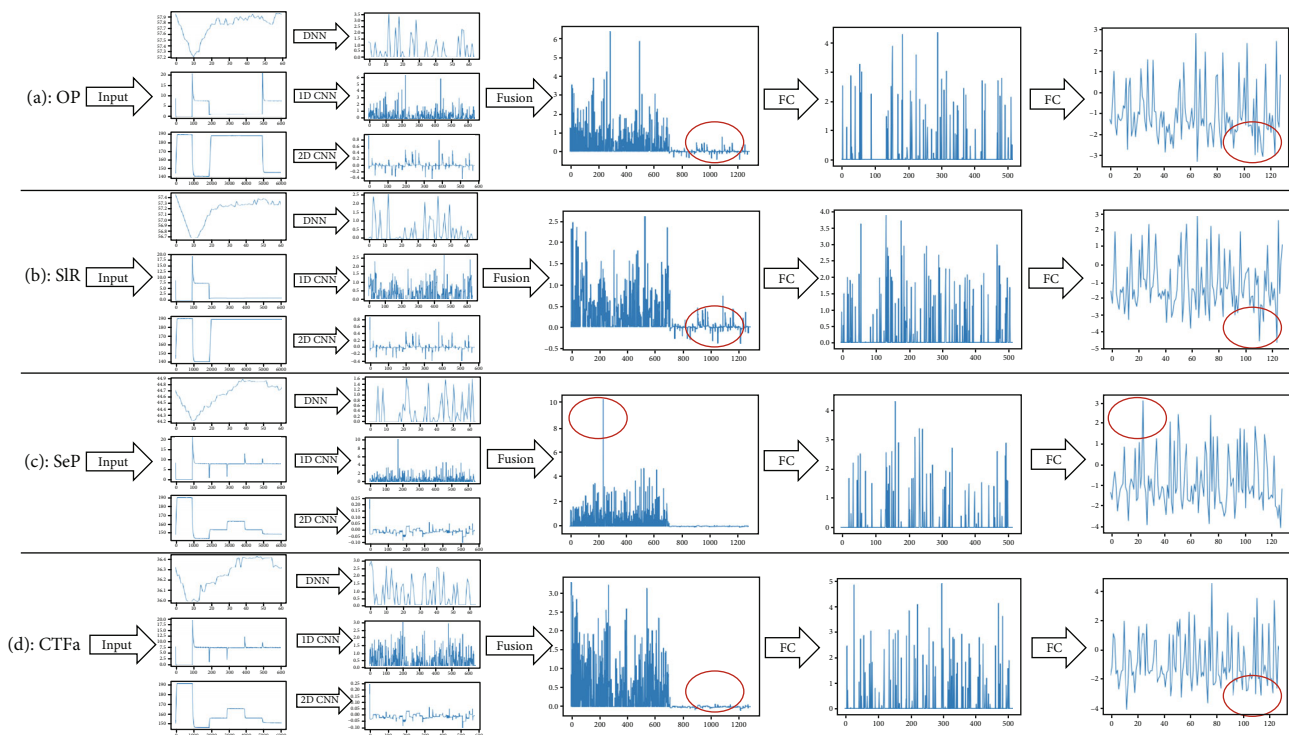

FIGURE 10: The whole evolution of signals of each healthy condition in MRSIFF.

convolutional layer to form the compound convolutional block, while maintaining the global and local features and improving the network capacity. In this structure, the multidimensional convolution layer and a full connection layer are used to extract features from multiscale input signals. In this method, the extracted features are concatenated to improve the accuracy of fault diagnosis, since the receptive fields between 1D convolution and 2D convolution are different.

The proposed MRSIFS, which is put forward, is tested on the hydraulic system condition monitoring dataset. The experiment results show that the fault diagnosis performance of the model based on 1D CNN is better than the model based on 2D CNN. However, when 1D CNN, 2D CNN, and DNN are combined, the accuracy of MRSIFS for fault diagnosis is improved to 97-99.5%, which indicates that multidimensional convolutional blocks can learn complementary and rich fault features. In addition, the same parameters and structures are used in MSFTFI, PCNN, and FAC-CNN, and the results show that the performance of the proposed Multirate Sensor Information Fusion Strategy is much better than all existing models.

Compared with the traditional intelligent fault detection models, the proposed MRSIFS has better performance in both feature extraction and detection accuracy. Therefore, the proposed method outperforms the existing methods.

4.9. Feature Work. In the future, the normal samples are far more than the fault samples due to the fault that will cause damage to the mechanical system, and existing research methods assume that the original signal dataset has sufficient balanced sample types. Therefore, it is necessary to develop a fault diagnosis model combining Generative Adversarial Networks (GAN), which is of great significance for the practical engineering environment and is also the research direction of the author in the future.

5. Conclusions

This paper introduced a Multirate Sensor Information Fusion Strategy (MRSIFS). The proposed method is based on multidimensional convolution block and time-frequency analysis technology, which implement multichannel parallel fault feature extraction, and the features from raw signals with different sampling rates are used for fault diagnosis. The multirate sensor feature extraction is a novel multitask feature extraction unit using a multidimensional convolution block and Adam loss function, which significantly improves the feature extraction capability. Finally, the simulation platform's experimental results show that the proposed multitask model achieves higher diagnosis accuracy than the existing methods. Besides, the manual feature selection for each task is unnecessary in MRSIFS, which has the potential toward a general-purpose framework. The main conclusions can be listed as follows:

(1) The hierarchical learning structure of multiple convolution blocks can be used to learn advanced fault features effectively

(2) To obtain fault sensitive and complementary detection features, this paper proposes a multidimensional convolution feature extraction model to adapt signal sources at different sampling frequencies

(3) Another valuable characteristic of the proposed strategy is the ability to work directly with raw sensor data, thus providing an end-to-end model to perform feature extraction and classification simultaneously

Data Availability

The hydraulic system condition monitoring dataset is available from the UC Irvine Machine Learning Repository at https://archive.ics.uci.edu/ml/datasets/Condition +monitoring+of+hydraulic+system.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

References

[1] C. Sun, Z. Zhang, Z. He, Z. Shen, and B. Chen, "Manifold learning-based subspace distance for machinery damage assessment," *Mechanical System & Signal Processings*, vol. 70-71, pp. 637–649, 2016.

[2] Z. du, X. Chen, H. Zhang, and R. Yan, "Sparse feature identification based on union of redundant dictionary for wind turbine gearbox fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 10, pp. 6594–6605, 2015.

[3] G. Jiang, H. He, J. Yan, and P. Xie, "Multiscale convolutional neural networks for fault diagnosis of wind turbine gearbox," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 4, pp. 3196–3207, 2019.

[4] B. Muruganatham, M. A. Sanjith, B. Krishnakumar, and S. A. V. Satya Murty, "Roller element bearing fault diagnosis using singular spectrum analysis," *Mechanical System & Signal Processing*, vol. 35, no. 1-2, pp. 150–166, 2013.

[5] B. Cai, Y. Liu, Q. Fan et al., "Multi-source information fusion based fault diagnosis of ground-source heat pump using Bayesian network," *Applied Energy*, vol. 114, no. 2, pp. 1–9, 2014.

[6] P. Imperatore, G. Persichetti, G. Testa, and R. Bernini, "Continuous liquid level sensor based on coupled light diffusing fibers," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 26, no. 4, pp. 1–8, 2020.

[7] L. Fang, D. Cunefare, C. Wang, R. H. Guymer, S. Li, and S. Farsiu, "Automatic segmentation of nine retinal layer boundaries in OCT images of non-exudative AMD patients using deep learning and graph search," *Biomedical Optics Express*, vol. 8, no. 5, pp. 2732–2744, 2017.

[8] J. Heaton, "Ian Goodfellow, Yoshua Bengio, and Aaron Courville: deep learning," *Genetic Programming and Evolvable Machines*, vol. 19, no. 1-2, pp. 305–307, 2018.

[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[10] L. Wen, X. Li, L. Gao, and Y. Zhang, "A new convolutional neural network-based data-driven fault diagnosis method," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5990–5998, 2018.

[11] F. Jia, Y. Lei, N. Lu, and S. Xing, "Deep normalized convolutional neural network for imbalanced fault classification of machinery and its understanding via visualization," *Mechanical System and Signal Processing*, vol. 110, pp. 349–367, 2018.

[12] D. Peng, Z. Liu, H. Wang, Y. Qin, and L. Jia, "A novel deeper one-dimensional CNN with residual learning for fault diagnosis of wheelset bearings in high-speed trains," *IEEE Access*, vol. 7, pp. 10278–10293, 2019.

[13] Z. Q. Chen, C. Li, and R.-V. Sanchez, "Gearbox fault identification and classification with convolutional neural networks," *Shock and Vibration*, vol. 2015, Article ID 390134, 10 pages, 2015.

[14] W. Zhang, G. Peng, C. Li, Y. Chen, and Z. Zhang, "A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals," *Sensors*, vol. 17, no. 2, p. 425, 2017.

[15] X. Ding and Q. He, "Energy-fluctuated multiscale feature learning with deep ConvNet for intelligent spindle bearing fault diagnosis," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 8, pp. 1926–1935, 2017.

[16] M. J. Hasan, M. M. M. Islam, and J.-M. Kim, "Multi-sensor fusion-based time-frequency imaging and transfer learning for spherical tank crack diagnosis under variable pressure conditions," *Measurement*, vol. 168, p. 108478, 2021.

[17] X. Xu, Z. Tao, W. Ming, Q. An, and M. Chen, "Intelligent monitoring and diagnostics using a novel integrated model based on deep learning and multi-sensor feature fusion," *Measurement*, vol. 165, p. 108086, 2020.

[18] S. Li, H. Wang, L. Song, P. Wang, L. Cui, and T. Lin, "An adaptive data fusion strategy for fault diagnosis based on the convolutional neural network," *Measurement*, vol. 165, p. 108122, 2020.

[19] H. Tao, P. Wang, Y. Chen, V. Stojanovic, and H. Yang, "An unsupervised fault diagnosis method for rolling bearing using STFT and generative neural networks," *Journal of the Franklin Institute*, vol. 357, no. 11, pp. 7286–7307, 2020.

[20] N. Helwig, E. Pignanelli, and A. Schütze, "Condition monitoring of a complicated hydraulic system using multivariate statistics," in *2015 IEEE Internation37al Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, pp. 210–215, Pisa, Italy, 2015.

[21] J. Schmidhuber, "Deep learning in neural networks: an overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[22] H. Qiao, T. Wang, P. Wang, L. Zhang, and M. Xu, "An adaptive weighted multiscale convolutional neural network for rotating machinery fault diagnosis under variable operating conditions," *IEEE Access*, vol. 7, pp. 118954–118964, 2019.

[23] H. Shao, H. Jiang, H. Zhang, and T. Liang, "Electric locomotive bearing fault diagnosis using a novel convolutional deep belief network," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 3, pp. 2727–2736, 2018.

[24] J. Wu, Y. Su, Y. Cheng, X. Shao, C. Deng, and C. Liu, "Multi-sensor information fusion for remaining useful life prediction of machining tools by adaptive network based fuzzy inference system," *Applied Soft Computing*, vol. 68, pp. 13–23, 2018.

[25] X. Ma, P. Wang, B. Zhang, and M. Sun, "An improved multi-channels information fusion model with multi-scale signals for fault diagnosis," in *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, pp. 552–555, Nanchang, China, March 2021.

[26] E. Jove, J. L. Casteleiro-Roca, H. Quintián, D. Simić, J. A. Méndez-Pérez, and J. Luis Calvo-Rolle, "Anomaly detection based on one-class intelligent techniques over a control level plant," *Logic Journal of IGPL*, vol. 28, no. 4, pp. 502–518, 2020.

[27] J. L. Calvo, S. H. Tindemans, and G. Strbac, "Risk-based method to secure power systems against cyber-physical faults with cascading impacts: a system protection scheme application," *Journal of Modern Power Systems and Clean Energy*, vol. 6, no. 5, pp. 930–943, 2018.

[28] R. Bernini, G. Persichetti, E. Catalano, L. Zeni, and A. Minardo, "Refractive index sensing by Brillouin scattering in side-polished optical fibers," *Optics Letters*, vol. 43, no. 10, pp. 2280–2283, 2018.

[29] E. Jove, J. L. Casteleiro-Roca, H. Quintián, J. A. Méndez-Pérez, and J. L. Calvo-Rolle, "A new method for anomaly detection based on non-convex boundaries with random two-dimensional projections," *Information Fusion*, vol. 65, pp. 50–57, 2020.

[30] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 936–944, Honolulu, HI, USA, 2017.

[31] N. Helwig, E. Pignanelli, and A. Schütze, "Condition monitoring of a complex hydraulic system using multivariate statistics," in *2015 IEEE Internation37al Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, pp. 210–215, Pisa, Italy, 2015.

[32] T. Schneider, N. Helwig, and A. Schütze, "Industrial condition monitoring with smart sensors using automated feature extraction and selection," *Measurement Science and Technology*, vol. 29, no. 9, p. 094002, 2018.

[33] J. Prakash and P. Kankar, "Health prediction of hydraulic cooling circuit using deep neural network with ensemble feature ranking technique," *Measurement*, vol. 151, p. 107225, 2020.

[34] B. Krishnapuram, M. Shah, A. Smola, C. Aggarwal, D. Shen, and R. Rastogi, "Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining," in *Acm Sigkdd International Conference on Knowledge Discovery & Data Mining ACM*, San Francisco California USA, 2016.

[35] M. Zhao, M. Kang, B. Tang, and M. Pecht, "Multiple wavelet coefficients fusion in deep residual networks for fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 6, pp. 4696–4706, 2019.

[36] Y. Li, Y. Chen, N. Wang, and Z. Zhang, "Scale-aware trident networks for object detection," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6053–6062, Seoul, Korea (South), 2019.

[37] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," 2015, https://arxiv.org/abs/1502.03167.

[38] D. Kingma and J. Ba, "Adam: a method for stochastic optimization," *Computer Science*, vol. 12, 2014.

[39] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[40] L. J. P. Van Der Maaten, "Accelerating t-SNE using tree-based algorithms," *The Journal of Machine Learning Research*, vol. 15, no. 93, pp. 3221–3245, 2014.

Research Article

Bearing Fault Diagnosis Based on Multiscale Convolutional Neural Network Using Data Augmentation

Seungmin Han, Seokju Oh, and Jongpil Jeong [ID]

Department of Smart Factory Convergence, Sungkyunkwan University, Suwon, Gyeonggi-do 16419, Republic of Korea

Correspondence should be addressed to Jongpil Jeong; jpjeong@skku.edu

Bearings are one of the most important parts of a rotating machine. Bearing failure can lead to mechanical failure, financial loss, and even personal injury. In recent years, various deep learning techniques have been used to diagnose bearing faults in rotating machines. However, deep learning technology has a data imbalance problem because it requires huge amounts of data. To solve this problem, we used data augmentation techniques. In addition, Convolutional Neural Network, one of the deep learning models, is a method capable of performing feature learning without prior knowledge. However, since conventional fault diagnosis based on CNN can only extract single-scale features, not only useful information may be lost but also domain shift problems may occur. In this paper, we proposed a Multiscale Convolutional Neural Network (MSCNN) to extract more powerful and differentiated features from raw signals. MSCNN can learn more powerful feature expression than conventional CNN through multiscale convolution operation and reduce the number of parameters and training time. The proposed model proved better results and validated the effectiveness of the model compared to 2D-CNN and 1D-CNN.

1. Introduction

The development of the IoT and industrial applications is rapidly improving the intelligence of equipment in the modern industry. As a result, mechanical equipment is becoming increasingly sophisticated and complex. Machinery failure can cause significant financial loss as well as human casualties. The rotating machine is one of the most widely used machines in the industry [1]. Rolling bearings are very essential components in rotating machines [2, 3]. Therefore, diagnosing bearing failure is very important.

Recently, data-based fault diagnosis [4–6] is drawing a lot of attention to the researchers due to the development of computers and GPUs. Traditional model-based diagnostic methods [7–11] are not efficient for learning nonlinear data. In addition, in the feature extraction step [12–15], there is a large difference in the result value depending on the skill of the expert. Machine learning methods such as Support Vector Machine (SVM) [16, 17], Principal Component Analysis (PCA) [18, 19], and artificial neural network (ANN) [20] have been used frequently. However, traditional machine learning [21–23] has also difficulty handling complex data.

On the other hand, data-driven diagnostics can effectively and accurately express the characteristics of big data or complex input data. With the advent of deep learning, it is possible to train neural networks through very deep continuous layers. Deep learning [24, 25] is widely applied in various fields such as image processing and image generation. DNN (Deep Neural Network) [26, 27] is a structure composed of many layers and can automatically extract deep features. Jia et al. [28] performed rolling bearing failure diagnosis via DNN and said that characteristics could be collected from raw signals. Xu et al. [29] used PCA to reduce the size of these features. Eren et al. [30] built an error detection system using 1D-CNN (One-Dimension Convolutional Neural Network). Deng et al. [31] proposed the MSIQDE algorithm based on making use of the merits of the Mexh wavelet function. Shao et al. [32] proposed the Convolutional Deep Faith Network (CDBN) for rolling bearings. Thus, deep learning technologies have the ability to overcome the shortcomings inherent in traditional machine learning methods.

Among deep learning models, CNNs are one of the great ways to perform feature learning without prior knowledge. CNNs are suitable for feature learning because they can pass

signals periodically. However, CNNs have some drawbacks. Unlike methods like SVM or PCA, it requires a lot of training samples. Also, the filter size of each convolutional layer is fixed, so you cannot get various information. In addition, since a CNN with a general structure can only extract single scale features, useful information may be lost and domain shift problems may occur. Therefore, we proposed an improved CNN called MSCNN with different filter sizes at each convolution. This allows us to extract useful information in frequency domains with different resolutions. In addition, more powerful feature expressions can be learned than conventional CNNs, and the number of parameters and training time can be reduced [33, 34]. Also, training deep learning requires a lot of data. If there is not enough training data, it is difficult to expect good results. Data on rolling bearings are not always enough under real condition. In this paper, we have increased the amount of data by applying permutation and time-warping techniques.

This paper is configured as follows: Neural networks and the background of the proposed model are described in Section 2. Experiment and results are provided in Section 3. Finally, Section 4 presents a conclusion.

2. Background

2.1. Artificial Neural Network (ANN).
The structure used in artificial neural networks [35] today was proposed by Frank Rosenblatt in 1958. Rosenblatt proposed a linear classifier called perceptron, which was a linear classifier structure that outputs 1 if the value is greater than 0 and -1 if it is less than 0 by adding the product of the inputs and weights and applying the activation function. Neurons have multiple inputs and one output. When each input is multiplied by a weight, the weight is multiplied by the next input, and the larger the weight, the more information is conveyed. A bias is added to the sum of the input value and the weight, and this bias represents the sensitivity of the neuron. Figure 1 describes the architecture of perceptron:

$$u = \sum_{i=1}^{n}(w_i \times x_i) + b, \tag{1}$$

where w represents the weight, x represents the input of neuron, and b represents the bias.

It was expected that the perceptron could create artificial intelligence like a real human, but in 1969, Minsky and Papert [36] proved the mathematical limitations of the perceptron, and people's expectations dwindled. According to them, the perceptron is only a simple linear classifier and cannot perform XOR classification. In other words, simple problems can be solved, but complex problems cannot be solved. Figure 2 describes the problem of OR, AND, and XOR.

In 1986, Rumelhart et al. [37] proposed a multilayered perceptron that overcomes the limitation of linear classifiers by adding a hidden layer. It proved that the XOR problem can be solved by using a concept of multilayered perceptron.
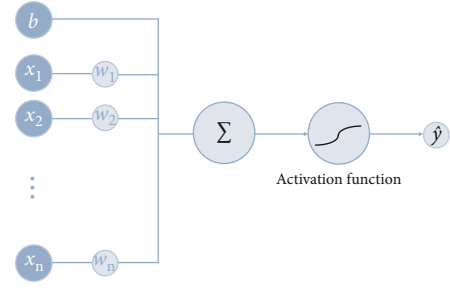

FIGURE 1: Architecture of perceptron.

MLP has a structure that is similar to a single perceptron, but by making the input/output characteristics of the intermediate layer and each unit nonlinear, it overcame the shortcomings of single perceptron by improving the network capability. In MLP, as the number of layers increases, the characteristics of decision regions formed by perceptron become more advanced. In Figure 3, we visualized the architecture of MLP.

2.2. Convolutional Neural Network (CNN).
CNN was developed by Lecun and Bengio [38] in the 1990s as a neural network structure that classifies handwritten numbers and received great attention. It is one of the most popular deep learning algorithms. CNN is a model that reduces the number of parameters using convolution using spatial relations. The goal of extracting hidden features from the data is to learn several feature filters in the input data and then perform operations between the feature filters and the input data. Since the vibration signal is a time series vibration signal, 1D-CNN [39] was used. CNN mainly consists of input, convolution layer, pooling layer, fully connected layer, and output. The basic structure of 1D-CNN is represented in Figure 4.

The convolutional layer is a layer that learns the feature values of the input data and consists of multiple feature maps. Neurons in each feature map are connected to the local area of the previous functional map through a set of weights. This set of weights is called the convolution kernel. The result after performing convolution on the input feature map and the convolution kernel is passed to the activation function to form the next feature map layer. Functional maps are computed through weight sharing, reducing model complexity, and making network training easier. The forward propagation of the convolutional layer is as follows:

$$x_j^n = f\left(\sum_{i \in F_j} x_i^{n-1} \times k_{ij}^n + b_j^n\right), \tag{2}$$

where x_j^l is the output of the layer l, M_j is the selected feature map, x_j^{l-1} is the output of the layer $l-1$, k_{ij}^l is the weight of layer l, and b_j^l is the bias of layer l.

The pooling layer is usually placed between successive convolutional layers. It is about reducing the dimensionality
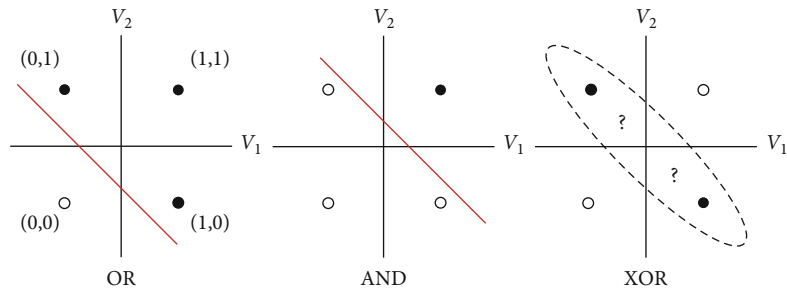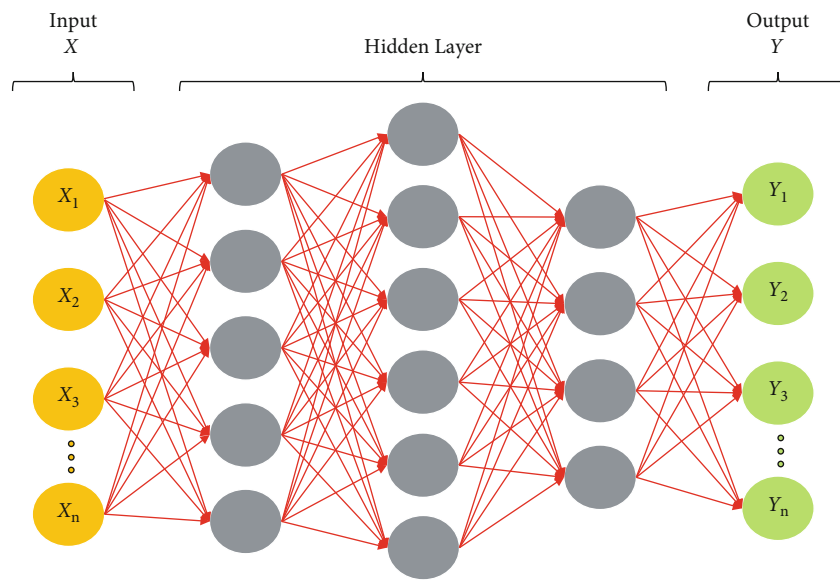
FIGURE 2: Problem of ANN.

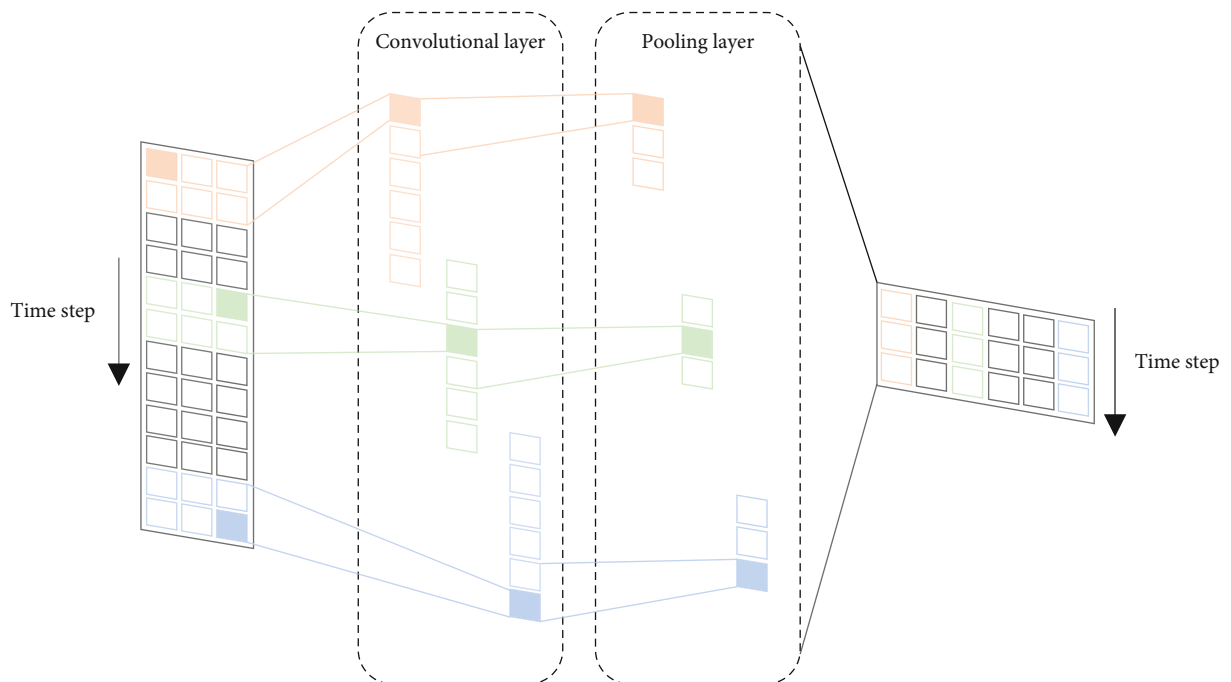FIGURE 3: Architecture of multilayered perceptron (MLP).

FIGURE 4: Architecture of One-Dimensional Convolutional Neural Network (1D-CNN).
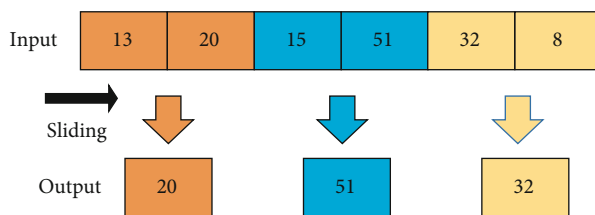
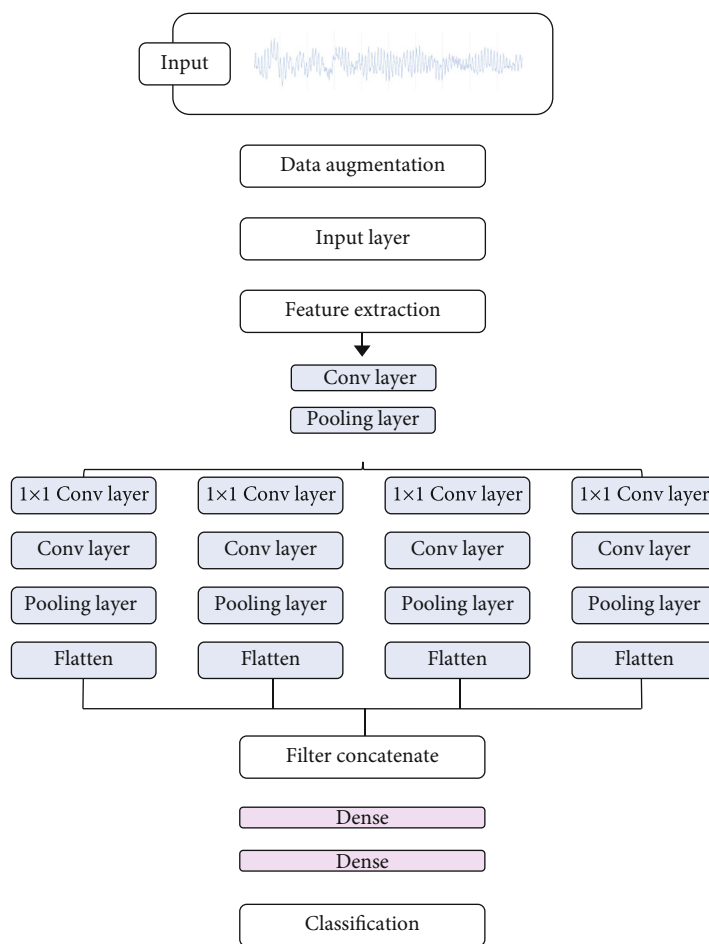FIGURE 5: Structure of maxpooling layer.
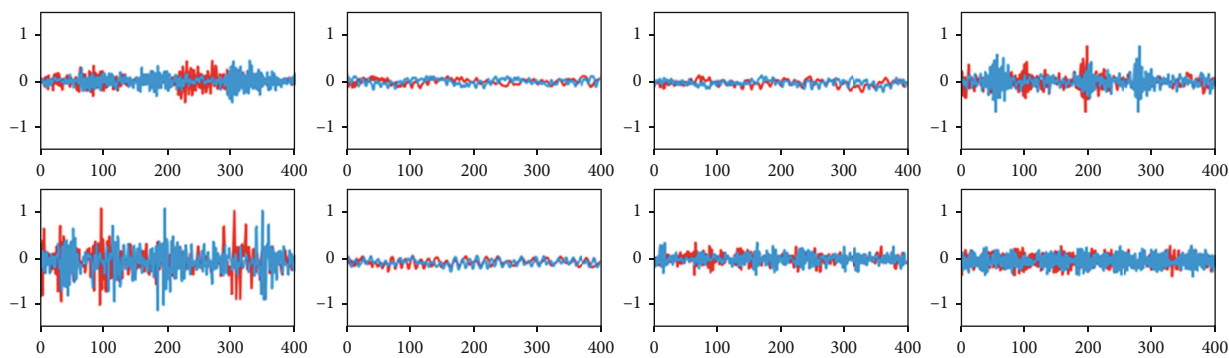

FIGURE 6: Framework of MSCNN.

FIGURE 7: Original signal (blue) and generated signal by time-warping (red).
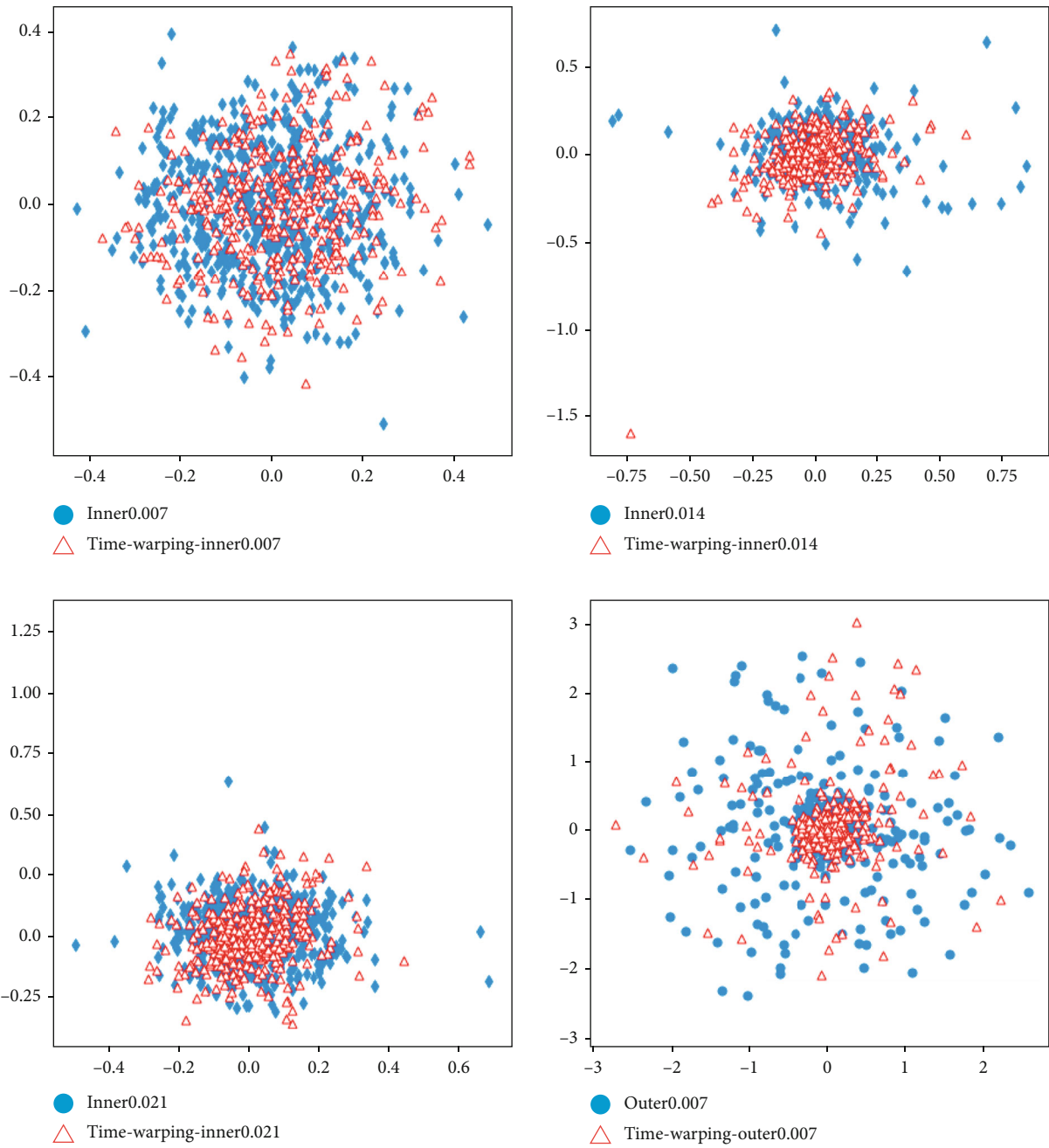
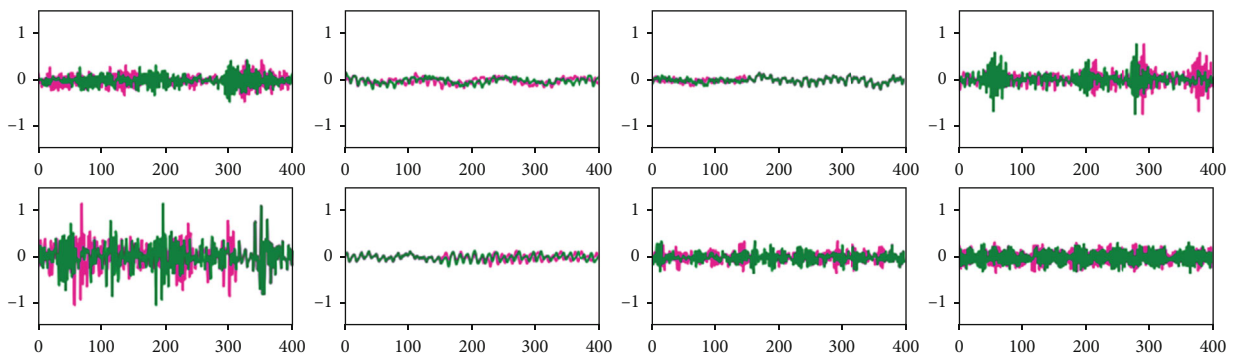Figure 8: Original signal (blue) and generated signal by time-warping (red).

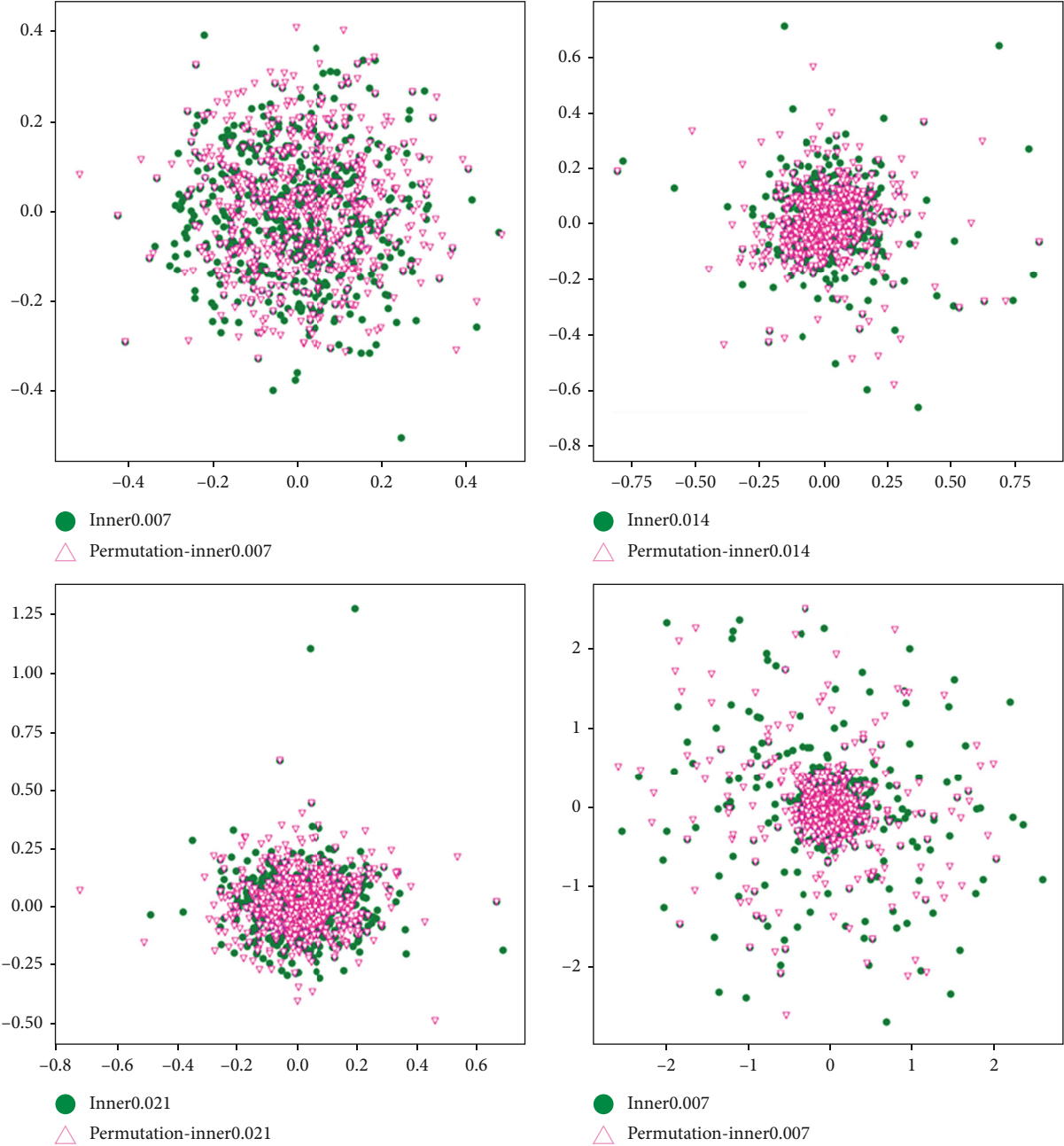Figure 9: Original signal (green) and generated signal by permutation (pink).

FIGURE 10: Original signal (green) and generated signal by permutation (pink).

TABLE 1: System specification.

Hardware environment	Software environment
CPU: Intel Core i7-8700 K, 3.7 GHz, six-core twelve threads, 16 GB	Windows TenserFlow 2.0 framework and Python 3.7
GPU: Geforce RTX 2080ti	

of the convolutional layer to do the extraction. The pooling layer uses feature vector values in feature maps for subsampling, so the most commonly used pooling methods are max-pooling and average pooling. In this paper, we used a maxpooling method that performs better in one-dimensional time series operations. The structure of the maximum pooling layer is shown in Figure 5 and is as follows:

$$p_i^{h_{n+1}}(j) = \max_{(j-1)w+1 \le t \le jw} \left(h_i^n(t) \right), \tag{3}$$
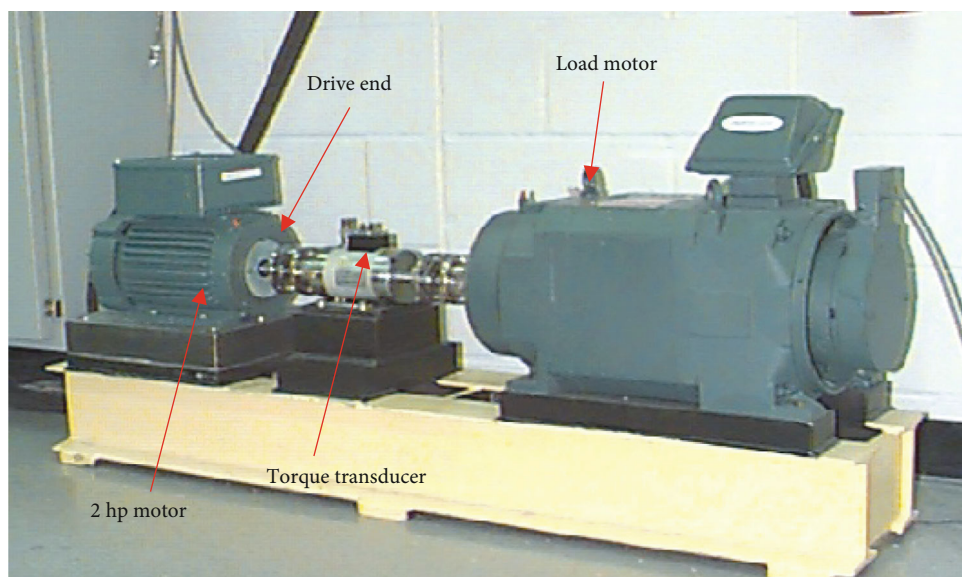
FIGURE 11: Bearing simulator of CWRU.

TABLE 2: Description of data—CWRU.

Training samples	Validation samples	Test samples	Fault types	Fault diameters	Class ID
1600	192	240	Normal	—	0
1600	192	240	Ball	0.007	1
1600	192	240	Ball	0.014	2
1600	192	240	Ball	0.021	3
1600	192	240	IR	0.007	4
1600	192	240	IR	0.014	5
1600	192	240	IR	0.021	6
1600	192	240	OR	0.007	7
1600	192	240	OR	0.014	8
1600	192	240	Outer	0.021	9

TABLE 3: Result of accuracy test—CWRU.

Model	MSCNN (%)	1D-CNN (%)	2D-CNN (%)	MSCNN (%) (no augmentation)
Train accuracy	99.88	99.31	99.27	98.56
Test accuracy	99.79	99.22	98.50	98.35

where q_i^l is the output of the tth neuron in the ith feature map of the layer l, $t \in [(j-1)W + 1, jW]$, W is the width of the pooled area, and $P_i^{l+1}(j)$ is the pooled value of the corresponding neuron in the layer $l + 1$.

2.3. Multiscale Convolutional Neural Network (MSCNN). Since the input value of CNN is usually a raw signal, poor results can be obtained regardless of hyperparameter changes if there is insufficient useful information. A con-

volution is the most important method to analyze the signal, and the size of the convolution filter in 1D-CNN has a great influence on the performance. In 1D-CNN, the size of the convolution filter is a hyperparameter. Since the convolution layer uses a convolution filter of a fixed size, laying out the size of the convolution filter is a very difficult problem.

Also, there are also some issues with the classification. First, a large size of convolutional filter has a good resolution because it focuses on a low-frequency region but tends to ignore high-frequency information. Conversely, a small size of convolutional filter focuses on the frequency band but has a lower resolution. Second, if a convolution filter of the same size is used, other discriminatory features cannot be properly extracted.

To address this problem, researchers proposed a multiscale convolution neural network. Multiscale convolution extracts features from a vibration signal using several convolution filters of various scales. Our framework of the proposed model is described in Figure 6.

We have used three convolution filters with different widths in the convolution layer to extract features from the original data through two convolution layers and one maxpooling layer and obtain three different feature maps and then concatenate them. The 1×1 convolution reduces the depth and width of the networks without increasing computation resources. This structure made it possible to extract other distinct features from the original signal. After concatenation, the classification part consisted of two fully connected layers and a softmax layer. The softmax function is designed as the number of labels in two different datasets.

The proposed multiscale feature extraction utilizes three convolution filters with different widths in the convolution layer to extract features from the original data through two convolution layers and one maximum
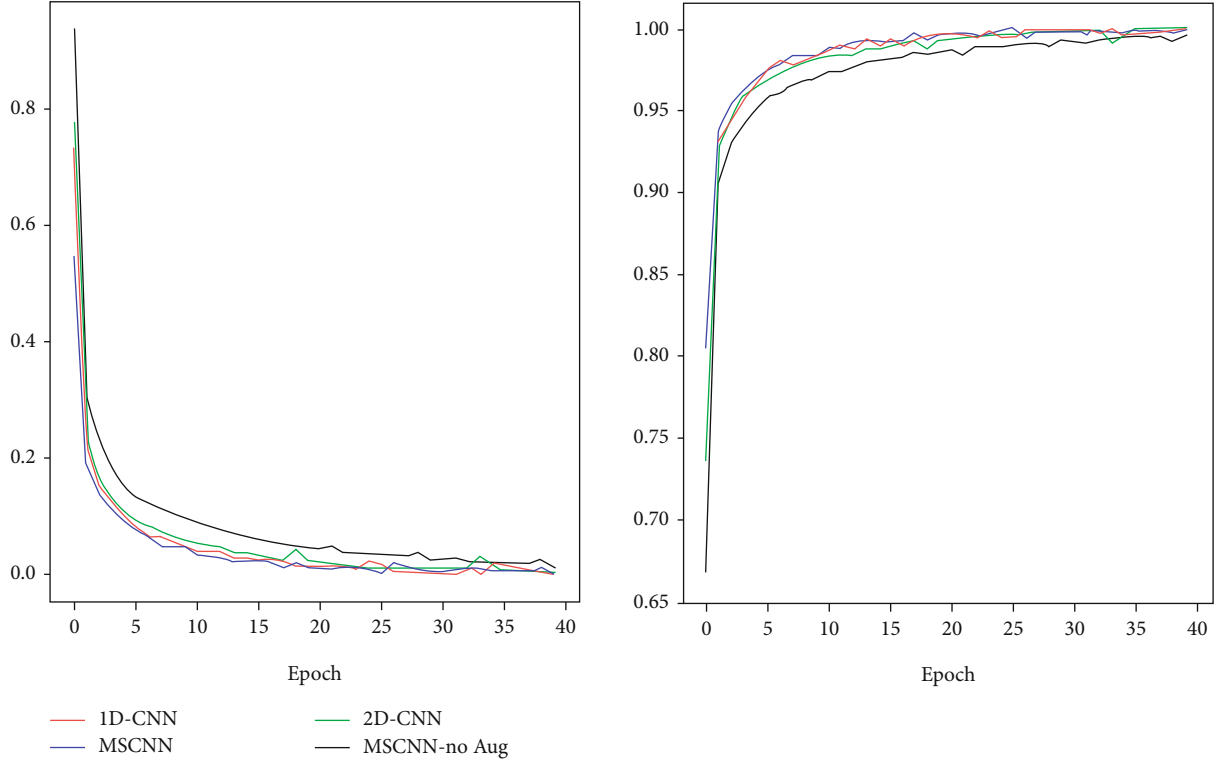
FIGURE 12: Loss and accuracy curves—CWRU.

pooling layer and obtain three different feature maps, then connect them. This structure allows both low- and high-frequency information to be obtained from the original signal. The convolution layer of MSCNN is as follows [40]:

$$C_d^t = \max\left(0, \text{Concat}\left(c_1^t, c_2^t, c_3^t\right)\right), \tag{4}$$

where C_d^t represents the output feature map of the tth convolution layer of MSCNN with depth $d = 3$ and c_1^t, c_2^t, c_3^t represents the feature maps after convolutions of MSCNN and can be seen as

$$\begin{cases} c_1^t = f_1^t * C^{[t-1]} + b_1^t, \\ c_2^t = f_2^t * C^{[t-1]} + b_2^t, \\ c_3^t = f_3^t * C^{[t-1]} + b_3^t. \end{cases} \tag{5}$$

Three convolution filters f_d^t are convolved with a $C^{[t-1]}$ feature map. b_d^t is the bias added to the feature map of the tth convolution layers. Each convolution layer is combined into a concatenation layer after convolution operations.

2.4. Data Augmentation. Deep learning increases the expressiveness of the model by increasing the parameters by stacking a lot of hidden layers. A huge amount of training data is necessary to properly train a lot of parameters. However, it is not easy to extract a lot of data under real working conditions. In addition, the data should keep the quality high and varied enough to reflect reality. If a deep learning model without enough training data is performed to train the parameters, the overfitting problem usually occurs. Therefore, by increasing the absolute amount of data even in a small data set area through a data augmentation technique [41, 42], we acquired new data by applying artificial changes to the data. Data augmentation can process unexplored input and improve the generalization effect of deep learning models. The important thing about data augmentation is to meet your domain knowledge to keep your existing labels when generating new data. It also does not change the data label with minor changes. Data augmentation technology is often used for images, but data augmentation technology is applied to time series data.

In this paper, we have used two data augmentation techniques. Both techniques are made by the fact that a slight change in the action point can keep the label. Firstly, a time-warping technique changes the position of time samples by smoothly distorting the time interval between samples. In Figures 7 and 8, we visualized the original signal and the generated signal by time-warping using subplot and scatter plot.

Second, the permutation is a technique that randomly changes the location of an event. To confuse the location of the input data, it is a technique to create a new window
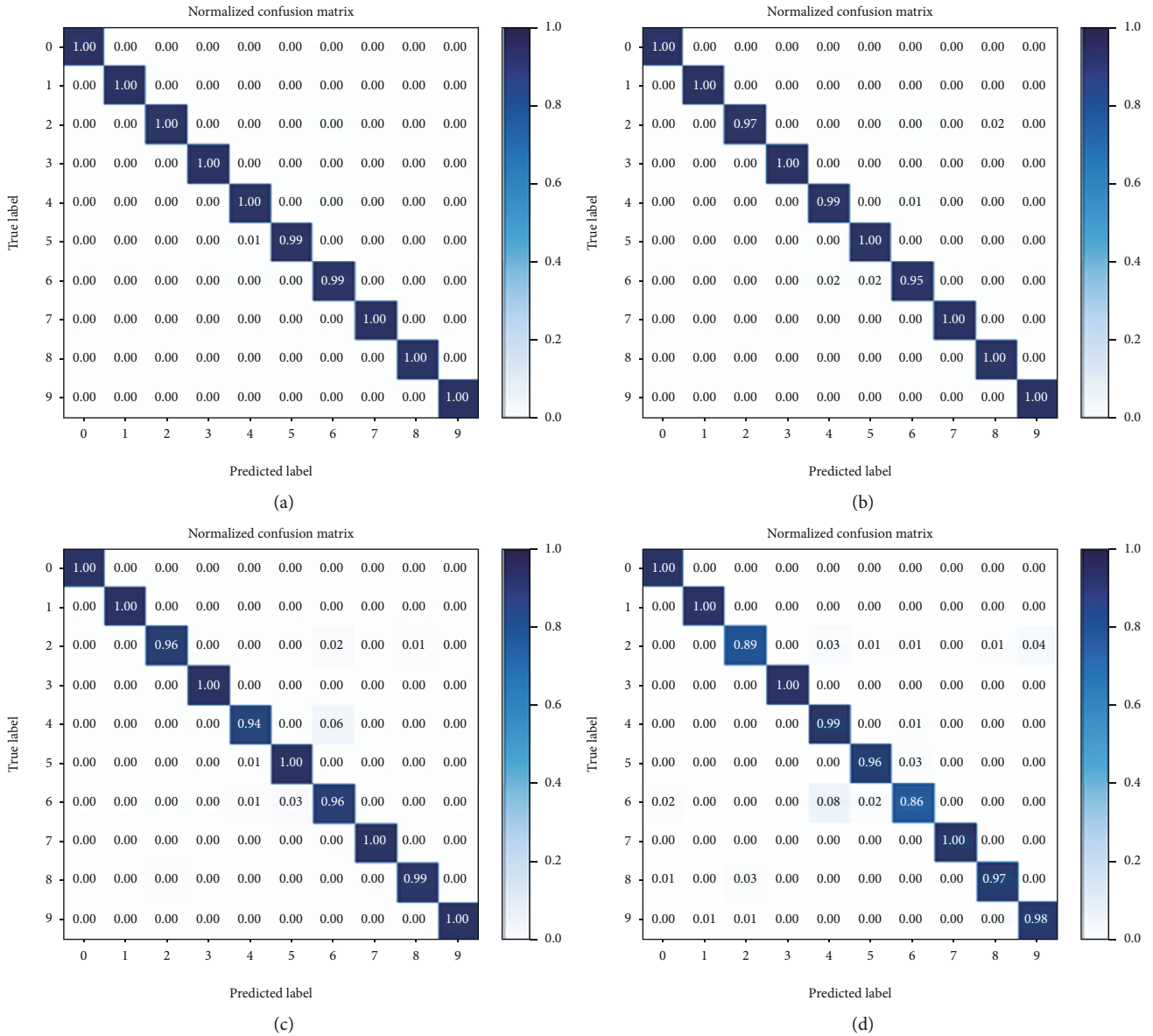
(a)

(b)

(c)

(d)

Figure 13: Confusion matrix of the proposed model (a), 1D-CNN (b), 2D-CNN (c), and proposed model without data augmentation (d)—CWRU.

by splitting the data into segments with the same length and then randomly changing the next segment. Figures 9 and 10 are diagrams visualizing the original data and the data generated by the permutation technique. Both methods have slight changes to the data, but no significant changes to the labels.

3. Experiment and Results

3.1. Experimental Configuration.
Keras library was used with Tensorflow backend. We compared our proposed model with 2 CNN models. The system specification for the experiment is shown in Table 1.

3.2. Simulation Case 1: CWRU Bearing Dataset.
To evaluate the performance of the proposed MSCNN, a bearing dataset from Case Western Reserve University Bearing Data Center was used, and the fault test bench is given in Figure 11.

The CWRU bearing dataset provides vibration signals generated by the simulator in normal and fail conditions. The 2 hp electric motor, torque converter, and dynamometer are the main components, and the vibration signals were collected from the drive end, fan end, and accelerometer mounted in the housing. We used motor drive end bearing data sampled at 0 hp, 1 hp, 2 hp, and 3 hp. Each defect type is divided into 0.007, 0.014, and 0.021 inches, so there are 10 states in the data set. We divided the size
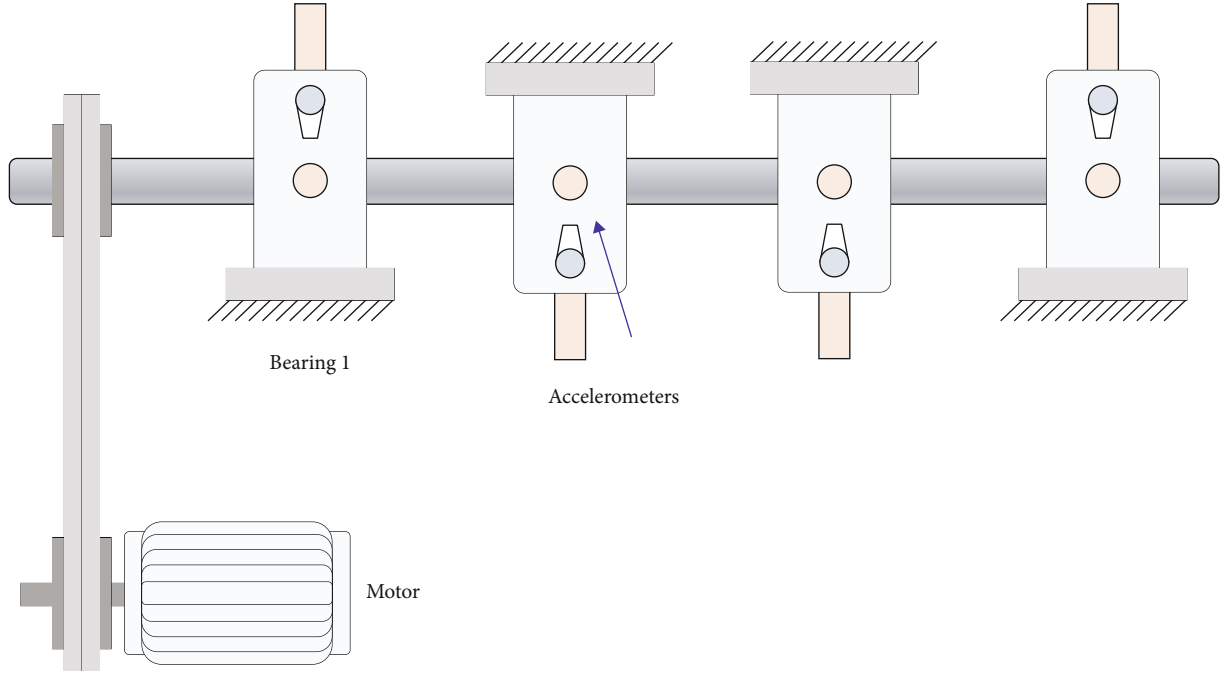
FIGURE 14: Bearing simulator of IMS.

TABLE 4: Description of data—IMS.

Training samples	Validation samples	Test samples	Fault types	Class ID
1500	192	240	Normal	0
1500	192	240	IR	1
1500	192	240	Roller element	2
1500	192	240	OR	3

of each signal segment by 400. Each state has 1600 samples, total of 10 states, so there are 16000 data. We divided 80% of the total data as training data, 20% as test data, and 20% of the remaining training data as validation data. The detailed description of data is in Table 2.

Two types of CNN models were compared to confirm the effectiveness of the proposed model. Both models are built as commonly used. Table 3 shows the results of an accuracy comparison of the proposed MSCNN with 1D-CNN, 2D-CNN, and the proposed model without increasing data.

Also, Figure 12 is a graph that visualizes the loss curve and accuracy curve for each model. In the loss curve, it can be seen that the proposed model is fine but settles more quickly. In addition, the accuracy curve shows that the proposed model is rapidly increasing in accuracy.

According to the result, the proposed model showed a better accuracy compared with others. The proposed model showed high accuracy of up to 1.4% and at least 0.53% in tests. There seems to be a slight difference, but this is significant because the 1% difference is not a small

number as the accuracy increases. In addition, there is a significant difference in accuracy even when data augmentation techniques are not used. This proves that splitting the convolution kernel into multiple scales works. In addition, we created a confusion matrix for each model for the reliability of the experiment.

As a second indicator, we used the confusion matrix. Although accuracy is the most intuitive metric, performance can be skewed on unbalanced datasets. There are four concepts you need to understand for the confusion matrix. The formula utilized for the confusion matrix is as below.

First, precision is the ratio of true to what the model classifies as true. Precision is expressed as

$$\text{Precision} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FP}|}. \tag{6}$$

Recall is the proportion of true that the predicted model is true. The recall is expressed as

$$\text{Recall} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}|}. \tag{7}$$

Accuracy is the most intuitive metric and is expressed as follows:

$$\text{Accuracy} = \frac{|\text{TP}| + |\text{TN}|}{|\text{TP}| + |\text{FP}| + |\text{FN}| + |\text{TN}|}. \tag{8}$$
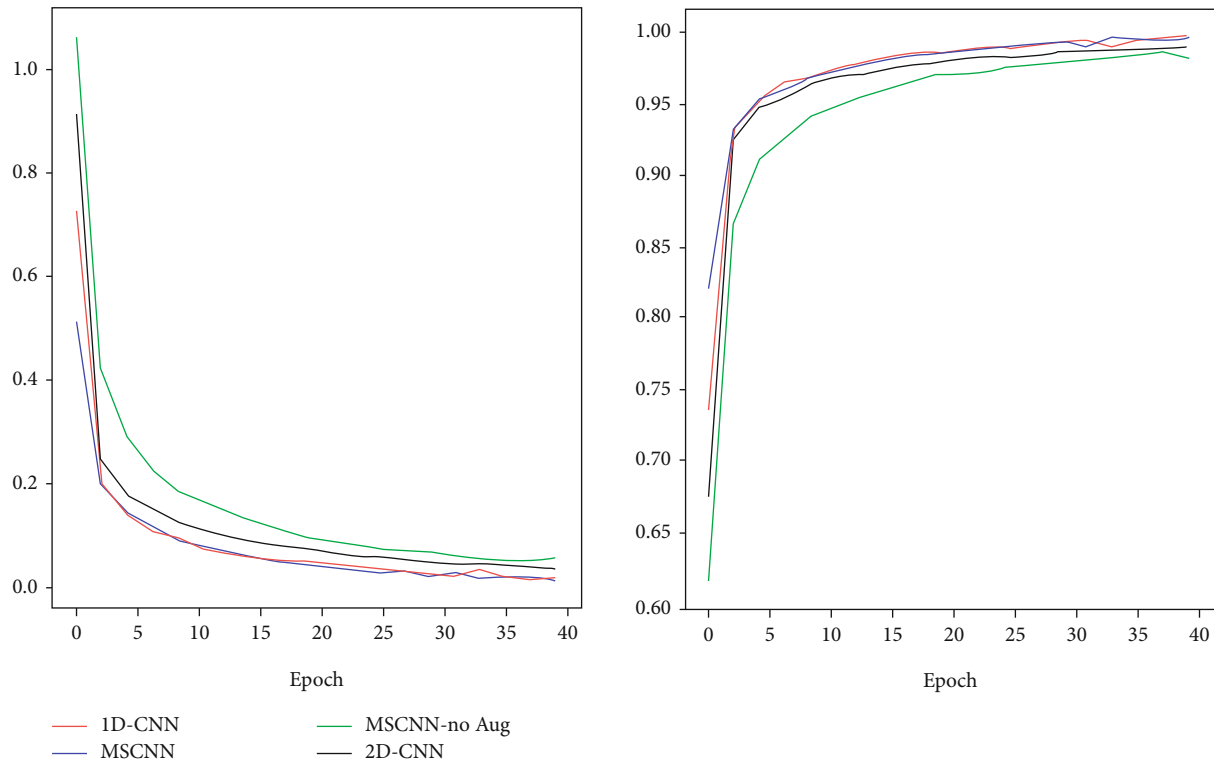
Figure 15: Loss and accuracy curves—IMS.

Table 5: Result of accuracy test—IMS.

Model	MSCNN (%)	1D-CNN (%)	2D-CNN (%)	MSCNN (%) (no augmentation)
Train accuracy	99.64	99.33	99.03	98.57
Test accuracy	99.51	99.25	99.01	98.14

The F1 score is the harmonic mean, which allows you to accurately evaluate the performance of your model if your data labels are unbalanced. The F1 score can be expressed as

$$\text{F1 score} = 2\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{9}$$

Figure 13 shows the results of the confusion matrix for each model. As a result of the experiment, as in the accuracy analysis, the proposed model showed the best performance, followed by 1D-CNN and 2D-CNN. Since the time series data is the input data, the performance of 1D-CNN appears to be better than that of 2D-CNN performance, followed by 1D-CNN and 2D-CNN. Since the time series data is the input data, the performance of 1D-CNN appears to be better than that of 2D-CNN.

3.3. Simulation Case 2: IMS Bearing Dataset. To add reliability to the evaluation of the proposed model, we used the bearing dataset provided by the Center for Intelligent Maintenance System (IMS), and the fault test bench is given in Figure 14.

Four rolling bearings were installed and operated with a rotational speed of 2000 rpm and a radial load of 6000 lbs. The raw signal was received by two accelerometers arranged vertically and horizontally. Therefore, there are a total of 4 types, including 3 fault states and normal states. In order to see the difference in the results, we conducted an experiment with the CWRU data set presented above and the number of samples made the same. The detailed description of data is in Table 4.

We compared the two types of CNN models and the proposed model as in the previous experiment. Table 5 shows the results of comparing the accuracy of the proposed model and a typical CNN model using IMS data. According to the results, the proposed model showed better accuracy compared to other models. The proposed model showed maximum 1.37%, minimum 0.26%, and higher accuracy in the test.

Also, Figure 15 is a graph that visualizes the loss curve and accuracy curve for each model. From the loss curve, it is shown that the proposed model is fine, similar to the CWRU experiment results, but settles more quickly. In addition, the accuracy curve shows that the proposed model is rapidly increasing in accuracy. The results of the confusion matrix for each model are shown in Figure 16.
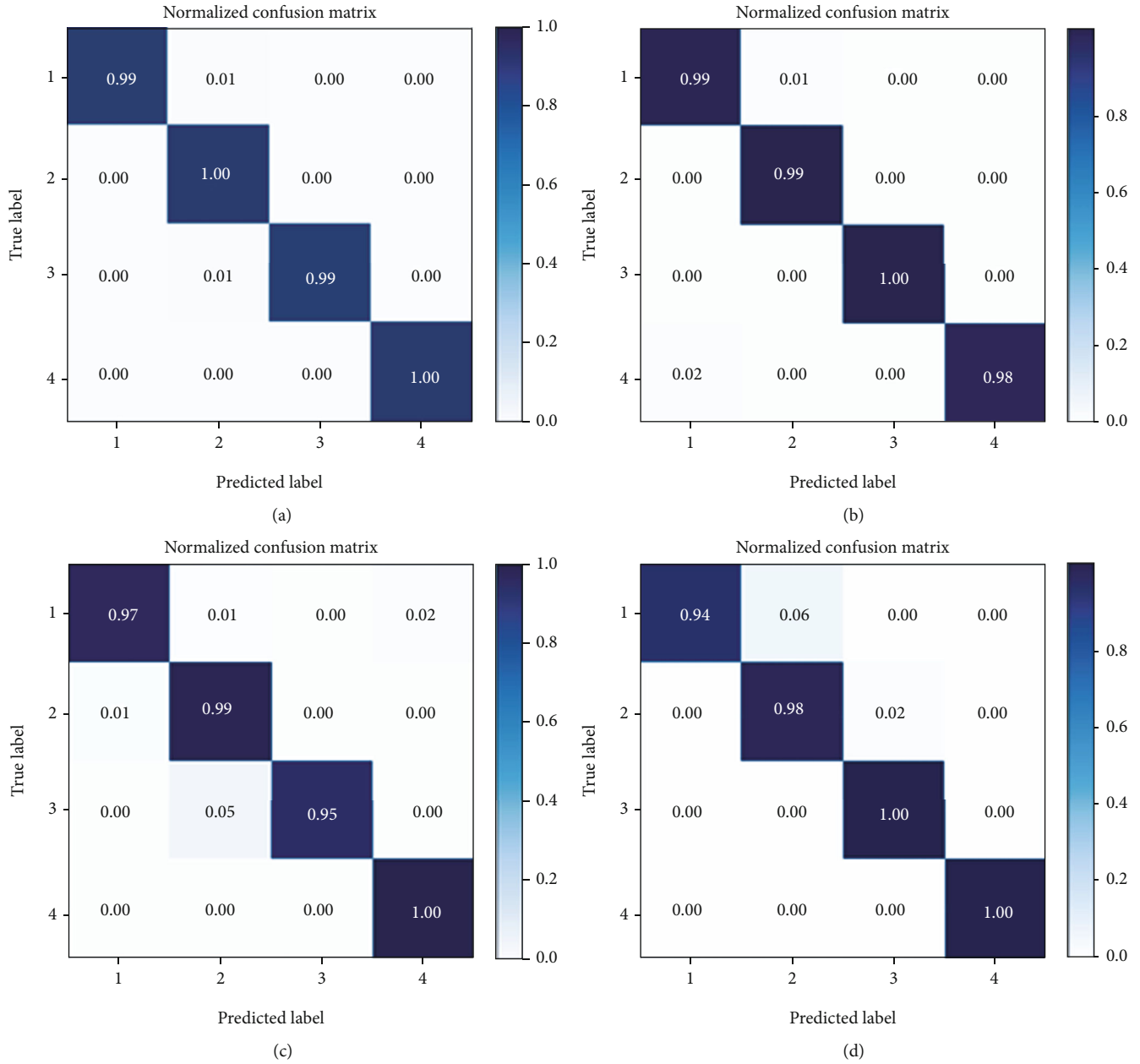
(a)

(b)

(c)

(d)

FIGURE 16: Confusion matrix of proposed model (a), 1D-CNN (b), 2D-CNN (c), and proposed model without data augmentation (d)—IMS.

4. Conclusion

Diagnosis of bearing failures is very important in the industry. Knowing bearing failure in advance can reduce downtime, prevent financial losses, and prevent failures in advance. Raw vibration signals were collected from CWRU and IMS bearing data sets. However, in the real world, there is not always enough data, and without enough data, the deep learning model performs very poorly. Therefore, in this paper, data was generated using data augmentation techniques that are good for application to two types of time series data. Experimentation has shown that data augmentation techniques significantly improve accuracy. In addition, in order to overcome the shortcomings of CNN, we proposed a model that minimizes the information lost through each different convolution filter by configuring the con-

volution layer in multiscale and reduces parameters and training time. The proposed model not only extracts useful information better from the frequency domain with different resolutions than the conventional CNN but also enables more powerful feature expression learning. The proposed model showed better performance than the existing 1D-CNN and 2D-CNN. Future research will apply to not only bearing data but also data from other fields widely used in the industry. In addition, we plan to improve the structure of the model more efficient.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request (jpjeong@skku.edu).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] Q. Gao, C. Duan, H. Fan, and Q. Meng, "Rotating machine fault diagnosis using empirical mode decomposition," *Mechanical Systems and Signal Processing*, vol. 22, no. 5, pp. 1072–1081, 2008.

[2] B. Li, M. Y. Chow, Y. Tipsuwan, and J. C. Hung, "Neural-network-based motor rolling bearing fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 5, pp. 1060–1069, 2000.

[3] Z. K. Peng, W. Tse Peter, and F. L. Chu, "A comparison study of improved Hilbert-Huang transform and wavelet transform: application to fault diagnosis for rolling bearing," *Mechanical Systems and Signal Processing*, vol. 19, no. 5, pp. 974–988, 2005.

[4] P. M. Frank, S. X. Ding, and T. Marcu, "Model-based fault diagnosis in technical processes," *Transactions of the Institute of Measurement and Control*, vol. 22, no. 1, pp. 57–101, 2016.

[5] S. Simani, C. Fantuzzi, and R. J. Patton, "Model-based fault diagnosis techniques," *Model-Based Fault Diagnosis in Dynamic Systems Using Identification Techniques*, pp. 19–60, 2003.

[6] W. Hongm, C. Tian-you, J. L. Ding, and M. Brown, "Data driven fault diagnosis and fault tolerant control: some advances and possible new directions," *Acta Automatica Sinica*, vol. 35, no. 6, pp. 739–747, 2009.

[7] S. X. Ding, *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*, Springer Science & Business Media, 2008.

[8] R. Isermann, "Model-based fault-detection and diagnosis - status and applications," *Annual Reviews in Control*, vol. 29, no. 1, pp. 71–85, 2005.

[9] P. M. Frank and X. Ding, "Frequency domain approach to optimally robust residual generation and evaluation for model-based fault diagnosis," *Automatica*, vol. 30, no. 5, pp. 789–804, 1994.

[10] A. K. Jalan and A. R. Mohanty, "Model based fault diagnosis of a rotor-bearing system for misalignment and unbalance under steady-state condition," *Journal of Sound and Vibration*, vol. 327, no. 3-5, pp. 604–622, 2009.

[11] R. Isermann, "Model-based fault detection and diagnosis - status and applications," *IFAC Proceedings Volumes*, vol. 37, no. 6, pp. 49–60, 2004.

[12] I. Guyon, S. Gunn, M. Nikravesh, and L. A. Zadeh, "Feature extraction," *Foundations and Applications*, vol. 207, 2008.

[13] H. Liu and H. Motoda, *Feature Extraction, Construction and Selection: A Data Mining Perspective*, vol. 453, Springer Science & Business Media, 1998.

[14] B.-C. Kuo and D. A. Landgrebe, "Nonparametric weighted feature extraction for classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 5, pp. 1096–1105, 2004.

[15] J. Mao and A. K. Jain, "Artificial neural networks for feature extraction and multivariate data projection," *IEEE Transactions on Neural Networks*, vol. 6, no. 2, pp. 296–317, 1995.

[16] A. Widodo and B.-S. Yang, "Support vector machine in machine condition monitoring and fault diagnosis," *Mechanical Systems and Signal Processing*, vol. 21, no. 6, pp. 2560–2574, 2007.

[17] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.

[18] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[19] R. Bro and A. K. Smilde, "Principal component analysis," *Analytical Methods*, vol. 6, no. 9, pp. 2812–2831, 2014.

[20] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*, MIT press, 1995.

[21] D. Michie, D. J. Spiegelhalter, and C. C. Taylor, "Machine learning," *Neural and Statistical Classification*, vol. 13, no. 1994, pp. 1–298, 1994.

[22] M. I. Jordan and T. M. Mitchell, "Machine learning: trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[23] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT press, 2012.

[24] I. Goodfellow, A. Courville, and Y. Bengio, *Deep Learning*, vol. 1, no. 2, 2016, MIT press, Cambridge, 2016.

[25] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[26] A. Canziani, A. Paszke, and E. Culurciello, *An Analysis of Deep Neural Network Models for Practical Applications*, 2016.

[27] W. Samek, A. Binder, G. Montavon, S. Lapuschkin, and K.-R. Müller, "Evaluating the visualization of what a deep neural network has learned," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 11, pp. 2660–2673, 2016.

[28] F. Jia, Y. Lei, J. Lin, X. Zhou, and N. Lu, "Deep neural networks: a promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data," *Mechanical Systems and Signal Processing*, vol. 72, pp. 303–315, 2016.

[29] F. Xu, G. Guohua, X. Kong, P. Wang, and K. Ren, "Object tracking based on two-dimensional PCA," *Optical Review*, vol. 23, no. 2, pp. 231–243, 2016.

[30] L. Eren, T. Ince, and S. Kiranyaz, "A generic intelligent bearing fault diagnosis system using compact adaptive 1D CNN classifier," *Journal of Signal Processing Systems*, vol. 91, no. 2, pp. 179–189, 2019.

[31] W. Deng, H. Liu, J. Xu, H. Zhao, and Y. Song, "An improved quantum-inspired differential evolution algorithm for deep belief network," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 10, pp. 7319–7327, 2020.

[32] H. Shao, H. Jiang, H. Zhang, and T. Liang, "Electric locomotive bearing fault diagnosis using a novel convolutional deep belief

network," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 3, pp. 2727–2736, 2017.

[33] Y. Song, D. Wu, W. Deng et al., "MPPCEDE: multi-population parallel co-evolutionary differential evolution for parameter optimization," *Energy Conversion and Management*, vol. 228, article 113661, 2020.

[34] W. Deng, J. Xu, Y. Song, and H. Zhao, "Differential evolution algorithm with wavelet basis function and optimal mutation strategy for complex optimization problem," *Applied Soft Computing*, vol. 100, article 106724, 2020.

[35] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain," *Psychological review*, vol. 65, no. 6, p. 386, 1958.

[36] M. Minsky and S. A. Papert, *Perceptrons: An Introduction to Computational Geometry*, MIT press, 2017.

[37] D. E. Rumelhart, G. E. Hinton, and J. L. McClelland, "A general framework for parallel distributed processing," *Parallel distributed processing: Explorations in the microstructure of cognition*, vol. 1, no. 26, pp. 45–76, 1986.

[38] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," *The Handbook of Brain Theory and Neural Networks*, vol. 3361, no. 10, p. 1995, 1995.

[39] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D. J. Inman, "Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks," *Journal of Sound and Vibration*, vol. 388, pp. 154–170, 2017.

[40] H. T. Mustafa, J. Yang, and M. Zareapoor, "Multi-scale convolutional neural network for multi-focus image fusion," *Image and Vision Computing*, vol. 85, pp. 26–35, 2019.

[41] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 279–283, 2017.

[42] T. T. Um, F. M. J. Pfister, D. Pichler et al., "Data augmentation of wearable sensor data for Parkinson's disease monitoring using convolutional neural networks," in *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pp. 216–220, ICMI '17: Proceedings of the 19th ACM International, 2017.

Research Article

Sylvester Matrix-Based Similarity Estimation Method for Automation of Defect Detection in Textile Fabrics

R. M. L. N. Kumari ⓘ,[1] G. A. C. T. Bandara ⓘ,[2] and Maheshi B. Dissanayake ⓘ[1]

[1]*Department of Electrical and Electronics Engineering, University of Peradeniya, KY 20400, Sri Lanka*
[2]*Department of Mechanical Engineering, University of Peradeniya, KY 20400, Sri Lanka*

Correspondence should be addressed to Maheshi B. Dissanayake; maheshid@eng.pdn.ac.lk

Fabric defect detection is a crucial quality control step in the textile manufacturing industry. In this article, a machine vision system based on the Sylvester Matrix-Based Similarity Method (SMBSM) is proposed to automate the defect detection process. The algorithm involves six phases, namely, resolution matching, image enhancement using Histogram Specification and Median–Mean-Based Sub-Image-Clipped Histogram Equalization, image registration through alignment and hysteresis process, image subtraction, edge detection, and fault detection by means of the rank of the Sylvester matrix. The experimental results demonstrate that the proposed method is robust and yields an accuracy of 93.4%, a precision of 95.8%, and computational speed of 2275 ms.

1. Introduction

Quality is an important aspect in the production line of the textile industry. Thus, fault detection in fabric quality control is an essential requirement of the textile industry. To minimize the manual labor in this endeavor, image analysis and processing techniques are widely used in the industry to automate the defect detection and classification process.

The defects that occur frequently on the fabric pattern limit the manufacturers who are able to recover only 45-65% of their profits from the off-quality goods [1, 2]. Hence, the defect detection process in the textile industry needs to satisfy high expectations of nearly 100% detection accuracy. Therefore, any other methods that are adopted should be able to perform real time defect detection with agility and accuracy. The main challenges encountered include the plethora of types and zones of defects to be detected, as well as the very fine variations that are present between the defects.

In many textile companies, the workers perform the fabric quality control process through human visual examination. As such, quality control is totally observer dependent, and it lacks uniformity. Further, the fabric quality control process is highly demanding for a human observer because the type of defects present will vary from fabric to fabric, according to the dynamic nature of the production process.

In this paper, we have generalized and automated fabric quality control, using the Sylvester Matrix-based defect detection algorithm, which can easily detect even very fine defects on fabrics, by comparing the input image with the reference image. Substantial literature sources are available, related to the algorithmic developments in the textile industry to detect defects [3–8]. As specified in [9, 10], an automated defect detection and classification system will certainly enhance the product quality, and result in heightened productivity. The autocorrelation method is one among the robust algorithms for detecting defects in both patterned and unpatterned fabrics [11]. Gabor Wavelet Network (GWN) was chosen as an effective technique to extract texture features from the textile fabrics. Depending upon the features extracted, an optimal Gabor filter was designed for defect detection [12–16]. Reference [17] presents the wavelet subwindow and gray level cooccurrence matrix for defect detection, and the Mahalanobis distance to categorize each wavelet subwindow as either defective or nondefective. Local
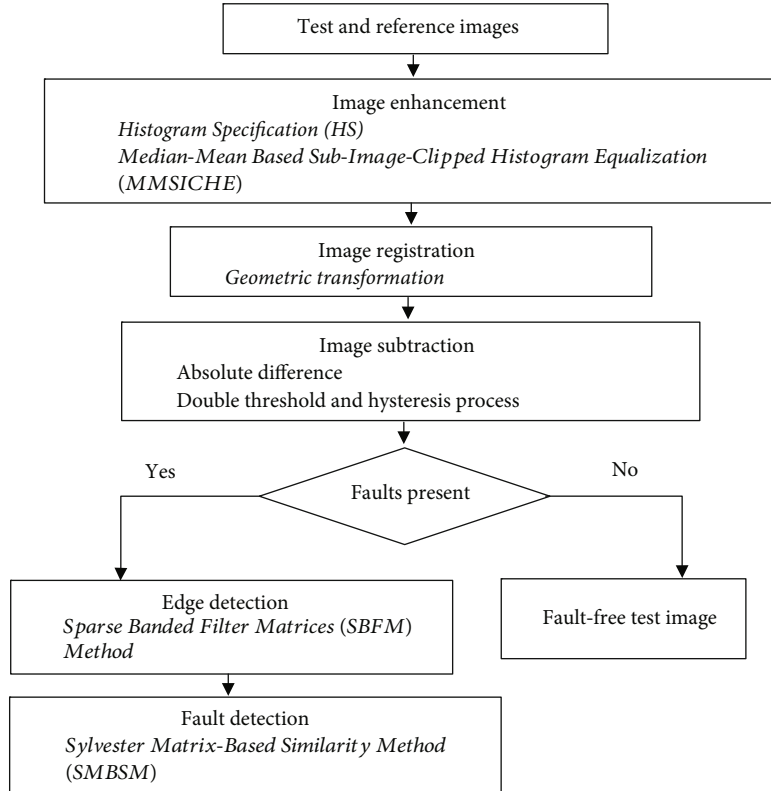
```
Test and reference images
        │
        ▼
Image enhancement
Histogram Specification (HS)
Median-Mean Based Sub-Image-Clipped Histogram Equalization
(MMSICHE)
        │
        ▼
Image registration
Geometric transformation
        │
        ▼
Image subtraction
Absolute difference
Double threshold and hysteresis process
        │
        ▼
   Faults present
   Yes ←          → No
        │              │
        ▼              ▼
Edge detection    Fault-free test image
Sparse Banded Filter Matrices (SBFM)
Method
        │
        ▼
Fault detection
Sylvester Matrix-Based Similarity Method
(SMBSM)
```

FIGURE 1: Block diagram of the proposed algorithm.

homogeneity and neural network-based defect detection algorithms are presented in [18]. A design that includes both hardware and software, and which uses the Otsu and Golden image subtraction methods, was proposed in [19] to reveal the defects. Its performance on a variety of defects validated the accuracy of the method developed.

In [20], the approach proposed was the fusion analysis for surface detection, which included a combination of the global and local features for the detection process by extracting and classifying the energy characteristics from the images. Based on the genetic elliptical Gabor filter, a novel method of defect detection was proposed in [21]. After being tuned by the genetic algorithm, the Gabor filter was applied to a variety of samples which show differences in type, shape, size, and background.

The Elo rating system was designed to inspect by making fair matches between the partitions from the images [22]. It was estimated to have 97% accuracy with the use of 336 patterned images. The particle analyzer method in [23] reveals higher performance compared to the other traditional methods, as it drives the analysis towards a predefined region of interest (ROI), and defines a particle as consisting of a minimum number of pixels. Moreover, a huge number of classes having large intraclass diversity continues to pose a major issue in the Feed Forward Neural Network (FFNN) and Support Vector Machine (SVM) dependent inspection methodologies, as all the classifiers require training regarding the known classes of fabric defects [24, 25].

The principal deficits present in the available literature are the overall lower accuracy and the substantial time for

making decisions. The methods described in [26, 27, 28] are able to achieve accuracy levels of only 90%, 90.6%, and 90.8%, respectively. The processing time of the algorithms revealed in [29, 30] stays high at 5.2 s and 5.9 s, respectively. Furthermore, the methods proposed in [31, 32] fail to give acceptably accurate performances, while detecting the finer defects in the fabrics.

In this paper, we describe a novel defect detection method which has fast processing and high accuracy, to detect even the very fine defects in the fabrics by comparing the reference and test images. In this method, all the images used are in RGB scale with identical resolution. First, image enhancement is done on every test image to ensure a better contrast image and thus facilitate defect detection. Later, image registration ensures that all the test images are in proper alignment. After this step, image subtraction is done to crosscheck the input against the reference image to detect any type of defects. If a positive rating is noted for the presence of defects, then edge detection is applied to both the reference and test images, to enable tracing even the finer details. Finally, the Sylvester Matrix-Based Similarity Method (SMBSM) is used to identify the defects in the fabrics. The method proposed works with 2275 ms computational speed and 93.4% average accuracy.

2. Proposed Methodology

In this research, an automated fault detection technique is proposed to lessen the degree of human interaction required for fault inspection in fabrics. Three types of fault inspection

algorithms exist, namely, the referential, nonreferential, and hybrid approaches [33]. The algorithm presented here is based on the referential approach, in which a reference image is employed to find defects in the test image. The proposed system is depicted in Figure 1.

It is well known that the performance of any image comparison algorithm is highly dependent upon the capturing condition of the input image. However, our system can analyze the images in the face of different capture conditions, in terms of contrast, distortion, and alignments, due to the image preprocessing techniques adopted. In the system proposed, the test image (I_{Test}) and the reference image (I_{Ref}) are in the RGB format with identical resolution.

2.1. Image Enhancement. Image enhancement aims at improving the quality of the test image captured under different lighting conditions. In the algorithm proposed, first the Histogram Specification (HS) improves the contrast level of the test image based on the reference image. Second, the Median–Mean-Based Sub-Image-Clipped Histogram Equalization (MMSICHE) algorithm was adopted as the processing technique to achieve the objective of preserving brightness, as well as image information content (entropy) together with control over the enhancement rate. This method circumvents excessive enhancement and provides images having natural enhancement, with the assurance that the test images taken under different lighting conditions will be accurately preprocessed to detect defects.

2.1.1. Histogram Specification (HS). The histogram specifications are used to rectify the contrast levels of the input test image against the reference image; i.e., if the contrast level of the input image is low in comparison to the reference, a correction will be applied to raise the contrast level and vice versa in the event of high contrast inputs [34].

The histogram of the intensity levels of both the reference and test images will be in the range of $[0, L - 1]$. The $n_{\text{Ref}}(i)$ and $n_{\text{Test}}(i)$ are the number of pixels having intensity i in the reference image and input test image, respectively, where $i = 0, 1, 2, \cdots, L - 1$. The inverse transformation, as defined in (1), maps i to $z(0 \leq z \leq L - 1)$ and shows the corresponding intensity values of the transformed test image ($G_{T,HS}$):

$$z = \left((L - 1) \sum_{z=0}^{i} \frac{n_{\text{Ref}}(z)}{p * q} \right)^{-1} \left((L - 1) \sum_{j=0}^{i} \frac{n_{\text{Test}}(j)}{p * q} \right), \quad (1)$$

where p and q are the row and column dimensions of the images, respectively.

2.1.2. Median–Mean-Based Sub-Image-Clipped Histogram Equalization (MMSICHE) for Contrast Enhancement. This method represents the MMSICHE algorithm which consists of three steps: median and mean calculation, histogram clipping, and histogram subdivision and equalization. MMSICHE further enhances the image quality of the transformed test image ($G_{T,HS}$).
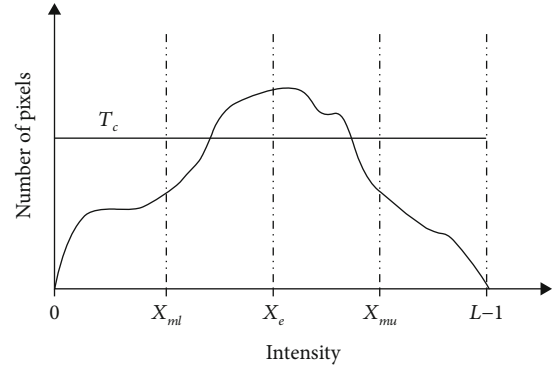

Figure 2: Process of histogram clipping.

The median of the image is shown to have an intensity value X_e, where the cumulative density function is around 0.5 [35]. Based on the median value, two mean intensity values, the mean of the lower histogram X_{ml} and the mean of the upper histogram X_{mu}, are calculated for two individual subhistograms. The corresponding values for X_e, X_{ml}, and X_{mu}, as shown in Figure 2, are calculated as in [35, 36] before the histogram clipping process.

Histogram clipping is done to control the degree of enhancement, to ensure that the resultant image is natural in appearance, matching that of the input image, as close as possible. The clipping threshold (T_c) is calculated as in [28, 35].

The image histogram is divided equally into four bins as shown in Figure 2. The subdivision process produces the four subimages W_{Ll}, W_{Lu}, W_{Ul}, and W_{Uu} ranging from the gray level 0 to X_{ml}, $X_{ml} + 1$ to X_e, $X_e + 1$ to X_{mu}, and $X_{mu} + 1$ to $L - 1$, respectively. In the next step of MMSICHE, based on the pixel distribution, all the four subhistograms are equalized individually and independently, using either (2), (3), (4), or (5) for the independent fine tuning:

$$F_{Ll} = X_{ml} \sum_{i=0}^{X_{ml}} \frac{H_c(i)}{N_{Ll}} \quad \text{for } W_{Ll}, \quad (2)$$

$$F_{Lu} = (X_{ml} + 1) + (X_e - (X_{ml} + 1)) \sum_{i=X_{ml}+1}^{X_e} \frac{H_c(i)}{N_{Lu}} \quad \text{for } W_{Lu}, \quad (3)$$

$$F_{Ul} = (X_e + 1) + (X_{mu} - (X_e + 1)) \sum_{i=X_e+1}^{X_{mu}} \frac{H_c(i)}{N_{Ul}} \quad \text{for } W_{Ul}, \quad (4)$$

$$F_{Uu} = (X_{mu} + 1) + (L - 1 - (X_{mu} + 1)) \sum_{i=X_{mu}+1}^{L-1} \frac{H_c(i)}{N_{Uu}} \quad \text{for } W_{Uu}, \quad (5)$$

where $H_c(i)$ is the clipped histogram. N_{Ll}, N_{Lu}, N_{Ul}, and N_{Uu} are the total number of pixels in the subimages W_{Ll}, W_{Lu}, W_{Ul}, and W_{Uu}, respectively.

The final step is to integrate all the subimages, W_{Ll}, W_{Lu}, W_{Ul}, and W_{Uu}, into one complete image ($G_{T,MMSICHE}$) for further analysis.

2.2. Image Registration. Image registration aims at finding the best transformation, which will align both the reference and input images. More precisely, it is used to identify a correspondence function, or mapping, that takes each spatial coordinate from the reference image and returns the coordinate for the test image. The transformation adopted involves two stages, namely, the geometric transformation and image resampling.

The geometric transformation [37] adopted is expressed as follows:

$$
\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix},
\tag{6}
$$

where (x', y') is the point coordinate of the test image and (x, y) is the corresponding point coordinate of the reference image. The transformation used in (6) has six degrees of freedom (DOF), where t_x and t_y relate to the translation of the signals, and a_{11}, a_{12}, a_{21}, and a_{22}, are used to calculate the scaling and shearing between the two images.

With this transformation, a correspondence map is established between the pixels in the preprocessed test image ($G_{T,MMSICHE}$) and that of the gray scale reference image (G_R), and the registered test image ($G_{T,IR}$) is generated.

2.3. Image Subtraction. Image subtraction is done to obtain the differential mapping between the reference image (G_R) and the preprocessed test image ($G_{T,IR}$). As image subtraction aims at identifying the presence of defects in the input, it will produce a binary decision which will be "1" if defects are present, and "0" otherwise. If defects are detected, the fabric is then transferred to the edge detection and fault detection stages, where the exact location and details of these defects in the fabric are identified. In the case nothing is detected, the fabric is labeled "defect-free".

In the algorithm proposed, the absolute difference (AD) is calculated in a pixel-wise subtraction process, as shown:

$$
AD = \left| G_{T,IR}(x, y) - G_R(x, y) \right|.
\tag{7}
$$

The output of AD may include a few erroneous pixels due to the uncorrected noise or misalignment between the two images. The double threshold approach is thus defined as eliminating the nonrelevant pixels which belong to the area between $HT_{\min}$ and $HT_{\max}$, as shown in Figure 3. The hysteresis process is then performed, where a weak pixel is transformed into a strong one, if and only if at least one strong pixel is present within its neighborhood, as depicted in Figure 3.
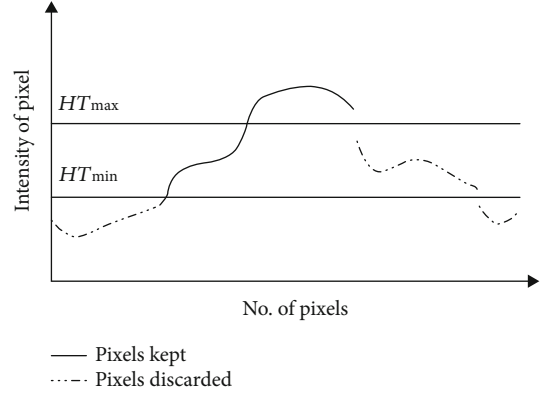
FIGURE 3: Hysteresis process.

2.4. Sparse Banded Filter Matrices (SBFM) for Edge Detection. In the algorithm proposed, Sparse Banded Filter Matrices (SBFM) [38] enables the detection of the edge information in both the test and reference images. SBFM comprises two major stages, namely, implementation of the zero-phase high-pass Butterworth filter using the SBFM matrix, and edge extraction. This edge detection method facilitates the finer details to be detected, of both the gray scale reference image (G_R) and registered test image ($G_{T,IR}$) significantly, thus ensuring the detection of even considerably insignificant defects on the test image during the fault detection stage.

Implementation of the zero-phase noncausal recursive high-pass filters based on banded matrices was introduced in [38] to identify the edge information from the images.

The matrix form of the first-order Butterworth high-pass filter can be expressed as follows:
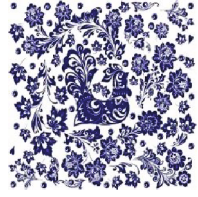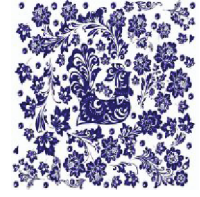
$$
y = A^{-1}Bx,
\tag{8}
$$

where A and B are the banded matrices of size $(N-1) \times (N-1)$ and $(N-1) \times N$, respectively, with N representing the length of the input signal. A and B are defined as follows:

$$
A = \begin{bmatrix}
a_0 & 0 & . & 0 & 0 & 0 \\
a_1 & a_0 & . & 0 & 0 & 0 \\
0 & a_1 & . & 0 & 0 & 0 \\
. & . & . & . & . & . \\
0 & 0 & . & a_1 & a_0 & 0 \\
0 & 0 & . & 0 & a_1 & a_0
\end{bmatrix},
$$

$$
B = \begin{bmatrix}
-1 & 1 & . & 0 & 0 & 0 & 0 \\
0 & -1 & . & 0 & 0 & 0 & 0 \\
0 & 0 & . & 0 & 0 & 0 & 0 \\
. & . & . & . & . & . & . \\
0 & 0 & . & 0 & -1 & 1 & 0 \\
0 & 0 & . & 0 & 0 & -1 & 1
\end{bmatrix}.
\tag{9}
$$

TABLE 1: The selected reference and test images of the dataset.

Image ID	Reference image (R)	Test image (T)
I1		
I2		
I3		

Furthermore, the transfer function of the zero-phase noncausal higher-order high-pass Butterworth filter can be expressed as follows:

$$h(z) = \frac{B(z)}{A(z)} = 1 - \frac{\alpha(-z + 2 - z^{-1})^d}{(-z + 2 - z^{-1})^d + \alpha(z + 2 + z^{-1})^d}, \quad (10)$$

where

$$\alpha = \left(\frac{(1 - \cos \omega_c)}{(1 + \cos \omega_c)}\right)^d, \quad (11)$$

d and ω_c are the filter order and cut-off frequency, respectively.

According to (10), the frequency response is maximally flat at $\omega = 0$, and the frequency response is of unity gain at $\omega = \pi$. Therefore, this is a zero-phase digital filter. The zero-phase high-pass Butterworth filter shown in (10) can be implemented using (8). Then, A and B can be defined as the banded sparse matrices of size $(N + 2d - 1) \times (N + 2d - 1)$ and $(N + 2d - 1) \times (N + 2d)$, respectively.

The sparse banded high-pass filter proposed is then applied row-wise and column-wise to extract the vertical and horizontal edges, respectively, as in [39], to detect all the edges of the test image processed ($G_{T,MMSICHE}$) and the gray scale reference image (G_R), while producing their corresponding edge extractions as $G_{T,SBFM}$ and $G_{R,SBFM}$, respectively.

2.5. Sylvester Matrix-Based Similarity Method (SMBSM) for Fault Detection. The Sylvester matrix (S) is associated with two univariate polynomials with the coefficients in a commutative ring [40]. This matrix helps to determine the common roots of the characteristic polynomial of the two images being compared. Hence, the similarity measure between the two images represents the rank or nullity of the matrix S.

$C(i, j)$ and $D(i, j)$ are 2D subimages of $G_{R,SBFM}$ and $G_{T,SBFM}$ such that $C(i, j), D(i, j) \in R^{n \times n}(n \le p, q)$ and square matrices. Their characteristic polynomials can be obtained by evaluating det $(\lambda I - C)$ and det $(\lambda I - D)$. These characteristic polynomials can be stated as follows:

$$P(C) = \sum_{i=0}^{n} c_i \lambda^{n-i}, \quad (12)$$

$$P(D) = \sum_{i=0}^{n} d_i \lambda^{n-i}. \quad (13)$$

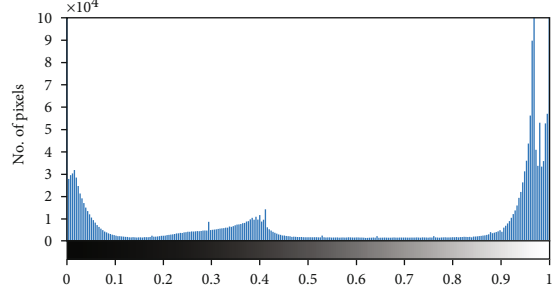The Sylvester matrix $S(E, F) \in R^{(n+n) \times (n+n)}$ of $P(C)$ and $P(D)$ can be defined as follows:
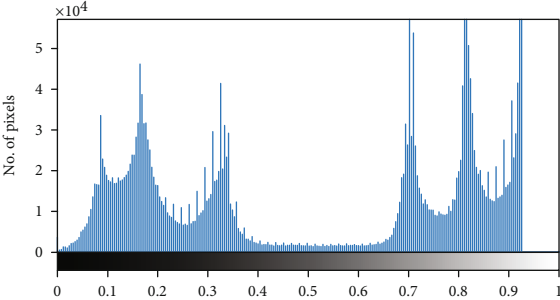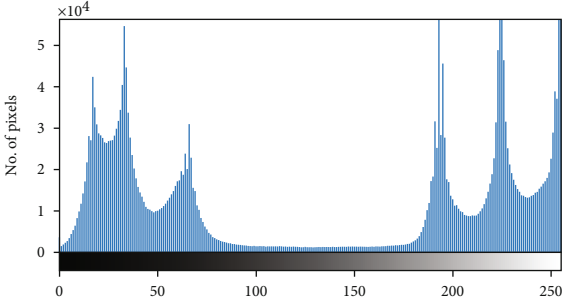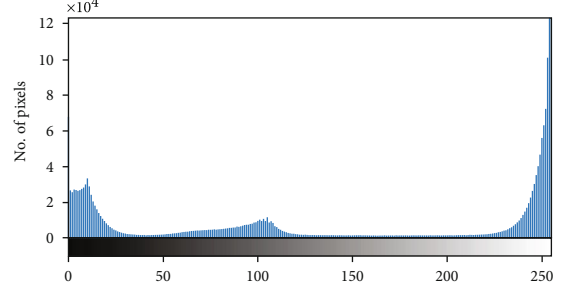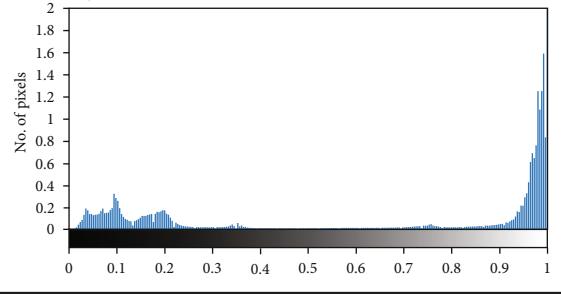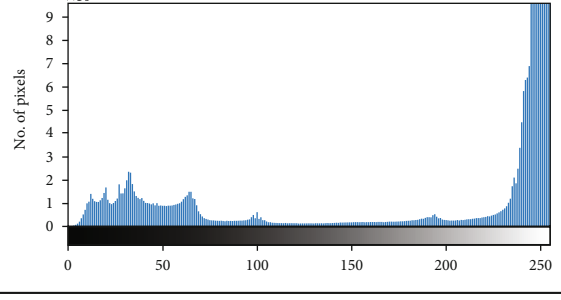
$$S(E, F) = [E(P(C)) \, F(P(D))], \quad (14)$$

where $E(P(C)) \in R^{n \times n}$ and $F(P(D)) \in R^{n \times n}$ are the Toeplitz matrices whose entries are the coefficients of $P(C)$ and $P(D)$, respectively [41] and can be defined as follows:

$$E(P(C)) = \begin{bmatrix} c_0 & 0 & . & 0 & 0 \\ c_1 & c_0 & . & 0 & 0 \\ . & c_1 & . & 0 & 0 \\ c_{n-1} & . & . & c_0 & 0 \\ c_n & c_{n-1} & . & c_1 & c_0 \\ 0 & c_n & . & . & c_1 \\ 0 & 0 & . & c_{n-1} & . \\ 0 & 0 & . & c_n & c_{n-1} \\ 0 & 0 & . & 0 & c_n \end{bmatrix},$$

$$F(P(D)) = \begin{bmatrix} d_0 & 0 & . & 0 & 0 \\ d_1 & d_0 & . & 0 & 0 \\ . & d_1 & . & 0 & 0 \\ d_{n-1} & . & . & d_0 & 0 \\ d_n & d_{n-1} & . & d_1 & d_0 \\ 0 & d_n & . & . & d_1 \\ 0 & 0 & . & d_{n-1} & . \\ 0 & 0 & . & d_n & d_{n-1} \\ 0 & 0 & . & 0 & d_n \end{bmatrix}. \quad (15)$$

TABLE 2: Histogram of test images after image enhancement.

Image ID	Histogram of test image before image enhancement	Histogram of test image after image enhancement
I1		
I2		
I3		

The nullity and rank of the matrix $S(E, F)$ show the degree of closeness of the characteristics of $P(C)$ and $P(D)$. For similar images, the nullity value $N(S(E, F))$ is equal to the number of columns in a matrix $S(E, F)$ and is zero for the totally dissimilar images. However, the value of rank, $r(S(E, F))$, is zero for similar images and equal to the number of the columns in a matrix $S(E, F)$ for those images that are totally dissimilar. In the event of small defects, the rank $r(S(E, F))$ too will be small, and the number is seen to rise as the defect intensity increases. Thus, rank $r(S(E, F))$ can be used in a defect intensity function to visualize the defective region, as it is in direct proportion to the defect intensity. Hence, to reach the final labeling in our method, we adopt the rank $r(S(E, F))$.

3. Results and Discussion
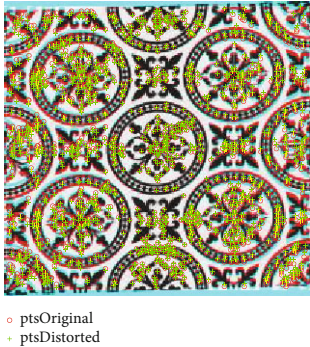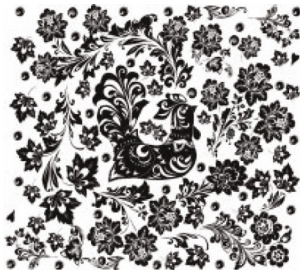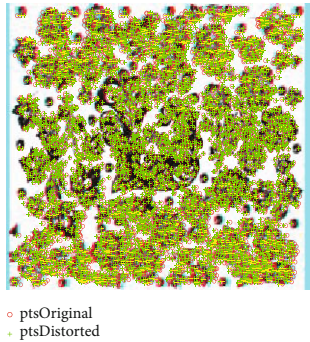
In this section, the findings of the simulation are presented. The algorithm proposed exhibits a significant improvement over the existing methods for defect detection in fabrics, as it is successful in identifying directional defects, under varying conditions of illumination.

The model was assessed in terms of robustness and stability using two datasets KTH-TIPS-I and KTH-TIPS-II [42]. The fabric dataset includes around 500 samples, captured under different conditions of illumination and contrast settings with skews, creating a challenge for defect detection. Table 1 lists three samples, including the reference (R) and test (T) images of the dataset. Further, right at the beginning, all the input images are resized to $1024 * 1024$.

First, the image enhancement method described in Section 2.1 is applied to enhance the histogram of the test image, as shown in Table 2. The intermediate outputs, post application of the image enhancement technique, are depicted in Table 3. Next, the image registration process presented in Section 2 is applied to align the test image coordinates with the corresponding reference image alignments, as revealed in Table 3. This preprocessing is done to circumvent any inaccuracies at the image subtraction stage, in which the double threshold values $HT_{\min} = 0.035$ and $HT_{\max} = 0.150$ are utilized, after manual tuning of the algorithm.

TABLE 3: Test images after image enhancement, coordinate matching and image registration.

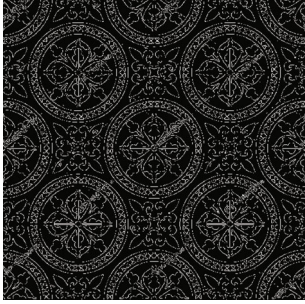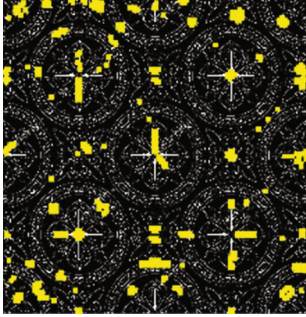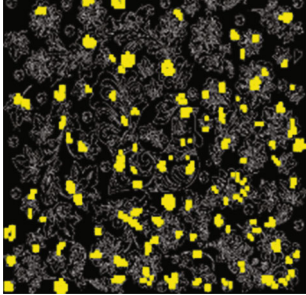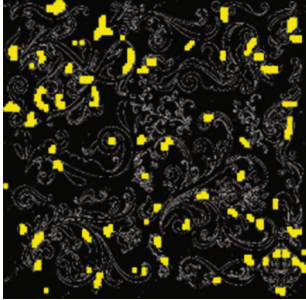| Image ID	Test image afterimage enhancement	Coordinates matching between reference and test image	After image registration
I1T		 ○ ptsOriginal + ptsDistorted	
I2T		 ○ ptsOriginal + ptsDistorted	
I3T		 ○ ptsOriginal + ptsDistorted	

In the textile industry, all the defects need to be detected with 100% accuracy. Keeping this objective in focus, the first priority is to minimize the rate of the false-positives which should be as negligible as possible. If false-negatives occur, then the fabric needs to be reinspected or discarded, even if the product is defect-free. In this experiment, the rates of the false-positives and false-negatives for the two datasets considered, KTH-TIPS-I and KTH-TIPS-II, are 4.2% and 0%, respectively. This occurred because of the carefully fine-tuned double threshold values used during the subtraction stage. After being subjected to image subtraction, all of the test images (I1T, I2T, and I3T) are identified as defective fabrics. Hence, these images are moved onto the next stage, namely, edge detection.

In the final stage of the algorithm, edge detection, as shown in Table 4, is performed using SBFM for the reference and test images, which are identified as positively defective, during the image subtraction process. The advantage of having this stage is to enhance all the minor details of the images, which will assist in detecting even the very fine defects present in the fabric.

The input parameters, selected by manual tuning, for the sparse banded high-pass filter design, including degree, cut-off frequency, and the length of the input sequence, are set as 3, 0.9, and 1024, respectively. The input image size of $1024 * 1024$ is subjected to zero padding, in order to match the sequence length (each row/column of the input image) given as the input to the filter design. From the analysis, it is clear that the edge extraction using SBFM provides more detailed results in the test and reference images, even the finer details, and the discontinuity in the edges extracted is less because the input parameters for this filter design are well tuned and matched. This is evident in the images, as revealed in Table 4.

TABLE 4: Edge detection of reference and test image, and fault detection of test image.

Image ID	Edge detection of reference image	Edge detection of test image	Fault detection on test image
I1			
I2			
I3			

A comparison of the similarity between the two images is performed using SMBSM. SMBSM is evaluated using window sizes of $4*4$, $8*8$, and $12*12$ pixels, with the minimum simulation time for the fault detection process being given by the $8*8$ window size. The reference and test images are first divided into small subwindows of $8*8$ and then compared against the coinciding location of the reference image. For each subwindow, the Sylvester matrix (S) is computed, and its rank is used to determine the defects on the selected subwindow of the test image compared to the reference image. The process is repeated for all the pixels of the entire image to detect faults, as shown in Table 4.

To calculate the accuracy of the algorithm that we proposed, the Binary Similarity Measure (γ) is used as given in (16), which detects the dissimilarities between two binary images based on a modified Hamming Distance measure [43]. The values of γ range from 0, distinct-dissimilarity, to 1, perfect similarity. The actual and detected faults on the test image are represented in the binary scale B_{AF} and B_{DF},
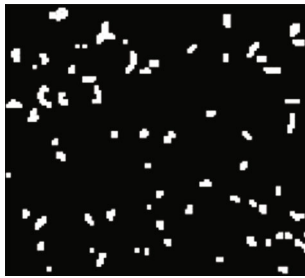
respectively, and the γ values of the test images are listed in Table 5.

$$\gamma = \left| 1 - \frac{2}{pq} \sum_{i=1}^{p} \sum_{j=1}^{q} (B_{AF}(i, j) \bigoplus B_{DF}(i, j)) \right|, \quad (16)$$

where the $\bigoplus$ symbol represents the logical exclusive-OR operator, and p and q are the row and column dimensions of the binary images of the actual and detected faults (B_{AF} and B_{DF}).

The computational speed of the algorithm proposed is about 2275 ms, making it superior to the existing ones. Furthermore, our algorithm performed with 93.4% accuracy, 95.8% precision, and 100% recall, on average. The proposed method works significantly well, even when the test images are taken under different conditions of illumination and have skews. The experiments presented here demonstrate the superiority of the proposed method.

TABLE 5: Actual and detected faults on the test image presented in binary scale.

Image ID	Actual faults on test image presented in binary scale (B_{AF})	Detected faults on test image presented in binary scale (B_{DF})	Binary similarity measure (γ)
I1			0.935
I2			0.932
I3			0.937

4. Conclusions

In this paper, a method to identify defects in fabrics has been proposed, based on the Sylvester Matrix-Based Similarity Method (SMBSM). This method is capable of handling misalignment and varying illuminations of the test images, captured under different conditions, as image enhancement improves the quality of the test image and image registration ensures proper alignment between the reference and test images. Edge detection is guaranteed to identify even very fine defects on fabrics during fault detection. Visual and quantitative results on the two datasets presented have demonstrated that the proposed method is superior and robust. In the future, more experiments will be conducted to further improve the accuracy of this method, and to assure that it is fast enough for defect detection in real time.

Data Availability

The dataset used in this research is freely available through "M. Fritz, B.C. E. Hayman, and J.O. Eklundh, THE KTH-TIPS Database. (Online) Available at: https://www.csc.kth.se/cvap/databases/kth-tips/download.html.

Conflicts of Interest

The authors declare that there is no conflict of interest.

References

[1] S. L. Bangare, N. B. Dhawas, V. S. Taware, S. K. Dighe, and P. S. Bagmare, "Fabric fault detection using image processing method," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 6, no. 4, pp. 405–409, 2017.

[2] J. Mahure and Y. C. Kulkarni, "Fabric faults processing: perfections and imperfections," *International Journal of Computer Networking, Wireless and Mobile Communications (IJCNWMC)*, vol. 4, no. 2, pp. 101–106, 2014.

[3] C. H. Chan and G. K. Pang, "Fabric defect detection by Fourier analysis," *IEEE Transactions on Industry Applications*, vol. 36, no. 5, pp. 1267–1276, 2000.

[4] Y. Hanand and P. Shi, "An adaptive level-selecting wavelet transform for texture defect detection," *Image and Vision Computing*, vol. 25, no. 8, pp. 1239–1248, 2007.

[5] D. M. Tsai and B. Hsiao, "Automatic surface inspection using wavelet reconstruction," *Pattern Recognition*, vol. 34, no. 6, pp. 1285–1305, 2001.

[6] X. Yang, G. Pang, and N. Yung, "Robust fabric defect detection and classification using multiple adaptive wavelets," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 152, no. 6, pp. 715–723, 2005.

[7] Y. X. Zhi, G. K. Pang, and N. H. C. Yung, "Fabric defect detection using adaptive wavelet," in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, Salt Lake City, UT, USA, May 2001.

[8] H. Y. T. Ngan, G. K. H. Pang, and N. H. C. Yung, "Automated fabric defect detection—a review," *Image and Vision Computing*, vol. 29, no. 7, pp. 442–458, 2011.

[9] C. S. Cho, B. M. Chung, and M. J. Park, "Development of real-time vision-based fabric inspection system," *IEEE Transactions on Industrial Electronics*, vol. 52, no. 4, pp. 1073–1079, 2005.

[10] R. S. Sabeenian, M. E. Paramasivam, and P. M. Dinesh, "Computer vision based defect detection and identification in handloom silk fabrics," *International Journal of Computer Applications*, vol. 42, no. 17, pp. 41–48, 2012.

[11] E. Hoseini, F. Farhadi, and F. Tajeripour, "Fabric defect detection using auto-correlation function," in *The 3rd International Conference on Machine Vision (IEEE Trans)*, pp. 557–561, Hong Kong, 2010.

[12] A. Kumar and G. Pang, "Defect detection in textured materials using Gabor filters," *IEEE Transactions Industry Applications*, vol. 38, no. 2, pp. 425–440, 2002.

[13] A. Bodnarova, M. Bennamoun, and S. Latham, "Optimal Gabor filters for textile flaw detection," *Pattern Recognition*, vol. 35, no. 12, pp. 2973–2991, 2002.

[14] D. M. Tsai and S. K. Wu, "Automated surface inspection using Gabor filters," *The International Journal of Advanced Manufacturing Technology*, vol. 16, pp. 474–482, 2000.

[15] G. Lambert and F. Bock, "Wavelet methods for texture defect detection," in *Proceedings of International Conference on Image Processing*, vol. 3, pp. 26–29, Santa Barbara, CA, USA, 1997.

[16] A. L. Amet, A. Ertuzun, and A. Ercil, "An efficient method for texture defect detection: sub-band domain co-occurrence matrices," *Image& Vision Computing*, vol. 18, no. 6-7, pp. 543–553, 2000.

[17] A. L. Amet, A. Ertüzün, and A. Erçil, "Texture defect detection using subband domain co-occurrence matrices," in *1998 IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 205–210, Tucson, AZ, USA, 1998.

[18] A. Rebhi, I. Benmhammed, S. Abid, and F. Fnaiech, "Fabric defect detection using local homogeneity analysis and neural network," *Journal of Photonics*, vol. 2015, Article ID 376163, 9 pages, 2015.

[19] P. Li, Z. Zhao, L. Zhang, H. Zhang, and J. Jing, "The real-time vision system for fabric defect detection with combined approach," in *8th International Conference on Image and Graphics*, pp. 13–16, Tianjin, China, August 2015.

[20] J. Guzaitis and A. Verikas, "Image analysis and information fusion-based defect detection in particleboards," *Electronics and Electrical Engineering*, vol. 71, no. 7, pp. 67–72, 2015.

[21] G. H. Hu, "Automated defect detection in textured surfaces using optimal elliptical Gabor filters," *Optik*, vol. 126, no. 14, pp. 1331–1340, 2015.

[22] C. S. Tsang, H. Y. Ngan, and G. K. Pang, "Fabric inspection based on the Elo rating method," *Pattern Recognition*, vol. 51, pp. 378–394, 2016.

[23] V. Tiwari and G. Sharma, "Automatic fabric fault detection using morphological operations on bit plane," *International Journal of Engineering Research & Technology (IJERT)*, vol. 2, no. 10, 2013.

[24] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital Image Processing Using MATLAB*, Gatesmark, USA, 2nd edition, 2009.

[25] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, Addison-Wesley Longman, Boston, USA, 1992.

[26] A. Khowaja and D. Nadir, "Automatic fabric fault detection using image processing," in *13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS)*, pp. 1–5, Karachi, Pakistan, 2019.

[27] J. S. Karthikeyan, "A survey on various fabric defect detection methods," in *2019 International Conference on Communication and Signal Processing (ICCSP)*, pp. 801–805, Chennai, India, 2019.

[28] S. Priya, T. Ashok Kumar, and V. Paul, "A novel approach to fabric defect detection using digital image processing," in *2011 International Conference on Signal Processing, Communication, Computing and Networking Technologies*, pp. 228–232, Thuckafay, India, 2011.

[29] N. Li, J. Zhao, and P. Jiang, "Fabric defects detection via visual attention mechanism," in *2017 Chinese Automation Congress (CAC)*, pp. 2956–2960, Jinan, China, 2017.

[30] L. Weninger, M. Kopaczka, and D. Merhof, "Defect detection in plain weave fabrics by yarn tracking and fully convolutional networks," in *2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, pp. 1–6, Houston, TX, 2018.

[31] A. Kumar, "Computer-vision-based fabric defect detection: a survey," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 1, pp. 348–363, 2008.

[32] D. Schneider, T. Holtermann, F. Neumann, A. Hehl, T. Aach, and T. Gries, "A vision-based system for high precision online fabric defect detection," in *7th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp. 1494–1499, Singapore, 2012.

[33] M. Moganti, F. Ercal, C. H. Dagli, and S. Tsunekawa, "Automatic PCB inspection algorithms: a survey," *Computer Vision and Image Understanding*, vol. 26, 1996.

[34] U. Khusanov and C. H. Lee, "Image enhancement based on local histogram specification," *Journal of Korean Institute of Intelligent Systems*, vol. 23, no. 1, pp. 18–23, 2013.

[35] Y. Wan, Q. Chen, and B. M. Zhang, "Image enhancement based on equal area dualistic sub-image histogram equalization method," *IEEE Transactions on Consumer Electronics*, vol. 45, pp. 68–75, 1999.

[36] T. Kim and J. Paik, "Adaptive contrast enhancement using gain-controllable clipped histogram equalization," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 4, pp. 1803–1810, 2008.

[37] R. C. Gonzales and R. E. Woods, *Digital Image Processing*, Pearson India, India, 4th edition, 2018.

[38] I. W. Selesnick, H. L. Graber, D. S. Pfeil, and R. L. Barbour, "Simultaneous low-pass filtering and total variation denoising," *IEEE Transactions on Signal Processing*, vol. 62, no. 5, pp. 1109–1124, 2014.

[39] V. Sowmya, N. Mohan, and K. P. Soman, "Edge detection using sparse banded filter matrices," *Procedia Computer Science*, vol. 58, pp. 10–17, 2015.

[40] J. R. Winkler and H. Halawani, "The Sylvester and Bézout resultant matrices for blind image deconvolution," *Journal of Mathematical Imaging and Vision*, vol. 60, no. 8, pp. 1284–1305, 2018.

[41] S. Barnett, *Polynomials and Linear Control Systems*, Marcel Dekker, New York, USA, 1983.

[42] M. Fritz, B. C. E. Hayman, and J. O. Eklundh, "THE KTH-TIPS Database," https://www.csc.kth.se/cvap/databases/kth-tips/download.html.

[43] A. A. Y. Mustafa, "A modified hamming distance measure for quick detection of dissimilar binary images," in *International Conference on Computer Vision and Image Analysis Applications*, pp. 1–6, Sousse, 2015.